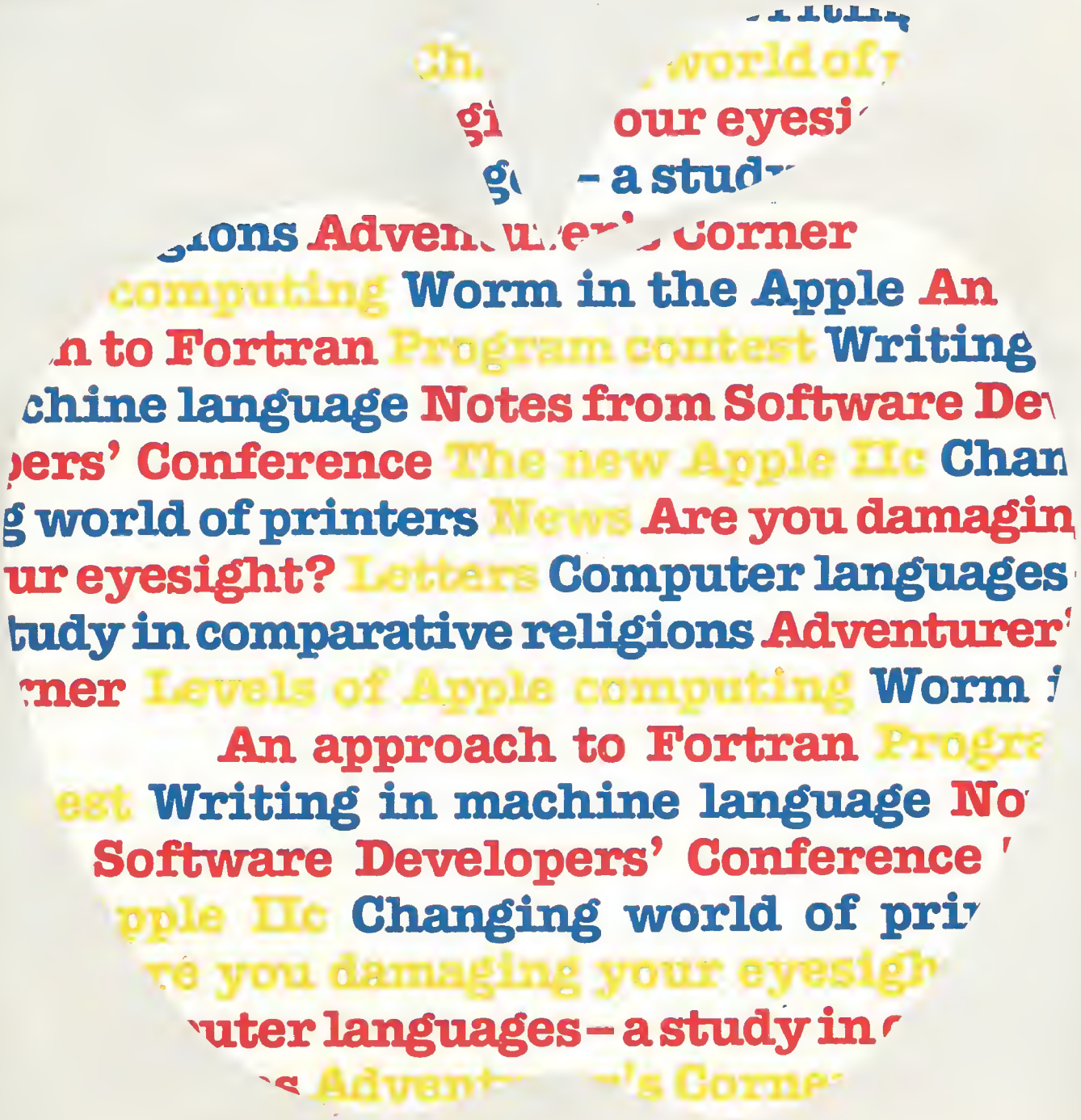
Registered by Australia Post Publication No NBG6313

The Australian Apple Review

The Independent Magazine

Vol 1 No 5 June $3*
* Recommended retail price



CompuMusic

103 ST JOHNS ROAD – GLEBE NSW 2037 – TELEPHONE 692 9875

APPLECARD ™

The next generation in CP/M® cards for your Apple® featuring extra speed and additional optional memory enabling it to act as a pseudodisc under DOS or CP/M.

6MHz 64K **$527.30**
6MHz 64K w/64K Extender **$720.00**
6MHz 64K w/128K Extender **$801.50**

80/88 CARD ™

This card upgrades the Apple to the speed, power and larger memory space of a 16-bit system. 64K and MBASIC are standard, CP/M 86 is also available.

80/88 Card **$801.50**

FLASHCARD

Disc emulator for Apple computers. Drive discs for DOS 3.3, CP/M & Pascal supplied with the card. Runs programs up to 10 times faster.

144K **$630.00**
288K $907.00

ULTRATERM ™

Revolutionary video display peripheral, features from 40-160 characters across and 24-48 lines.

Ultraterm **$470.00**

ULTRAPLAN ™

The complete spreadsheet program.

Ultraplan **$220.00**
w/Ultraterm **$615.00**

RANA SYSTEMS

Floppy disc drive w/652K unformatted. Uses DOS 3.3, CP/M & Pascal, PRODOS coming.

Elite Three **$950.00**

COLOUR FLOPPIES

Solid colour floppies, in 12 colours enabling rapid identification, colour coding and filing.

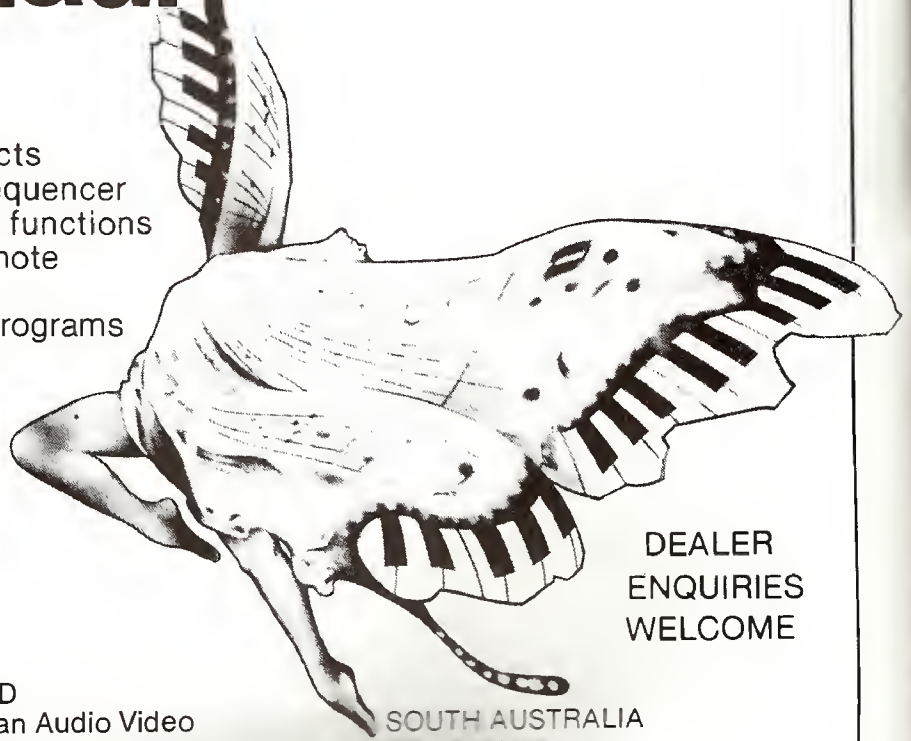
5¼" SS/DD, 40 TPI ..... **$ 5.50**
5¼" DS/DD, 40 TPI ..... **$ 7.20**

DEALER ENQUIRIES WELCOME

'the ultimate in affordable digital synthesizers'

alphaSyntauri™

A 16 oscillator, 8 voice polyphonic computer music system that can:
CREATE * sounds : envelopes : effects
PLAY * full polyphony : 3000 note sequencer
RECORD * 16 tracks : full multitrack functions
COMPOSE * play it and print it in 8 note polyphony.
EDUCATE * aural visual education programs for classroom and home



VICTORIA
Music Junction, 204 Camberwell Road
Hawthorn East 3123 Telephone: 82 7331

Brashs 108 Elizabeth Street
Melbourne 3000 Telephone: 63 6701

Troy Music 568 Chapel Street
South Yarra 3141 Telephone: 63 6701

DEALER ENQUIRIES WELCOME

WESTERN AUSTRALIA
Zenith Music
309 Sterling Hwy
Claremont 6010
Telephone: 383 1422

QUEENSLAND
North Australian Audio Video
336 Flinders Street
Townsville 4810
Telephone: (077) 72 4747

SOUTH AUSTRALIA
Allan's Music
Gawler Place
Telephone: 223 5533

The Australian Apple Review
Top Rear, 4 Carrington Rd,
Randwick NSW 2031
Phone: (02) 398 5111

Published by Gareth Powell Pty Ltd

Editor: Gareth Powell
Art Director: Avrille Goldsworthy
Production Managers:
Brenda Powell, Gary Williams
Photographer: Gary Williams
Subscription Manager:
Tina Spathos

Advertising Enquiries:
Craig Hoskinson, Patricia Barr
Gareth Powell Pty Ltd,
Top Rear 4 Carrington Rd,
Randwick NSW 2031
Phone: (02) 398 5111

Typesetting:
Printaction

Distribution:
GORDON & GOTCH LTD SYDNEY

Printed by: Ian Liddell Commercial Printing

Contents

As we go to press the news has come that the appeal by Apple against the "Wombat" decision has been successful. No one is yet in a position to make formal pronouncements on the probable results of this decision, but it seems very likely there will be no more copies of Apple brought into Australia, at least in commercial quantities.

This is further underlined by the statement by the Attorney-General, Gareth Evans, that new legislation was ready to be hustled through parliament to give copyright protection to programs incorporated in ROM.

Taking these two factors together, it can be assumed that the days of the "el cheapo" Apple are over.

Anybody buying a fake Apple at this stage would be putting themselves at great risk, because the freezing up of supply of fake Apples also means the freezing up of the supply of spare parts.

It is important to note that this will not in any way affect peripherals for adding on to genuine Apples. Owners will still have a wide range of printers, plotters, screens, boards and modifications that will not in any way upset the Applecart.

Indeed, Apple have shown themselves most willing to encourage manufacturers and suppliers of such peripherals.

It also means the end of Software Liberation, as people selling broken software will do so at their peril. It is interesting that the actions taken in Australia have been almost exactly duplicated by similar legal action in the United States of America and, to a much lesser extent, Taiwan.

If you have a fake Apple and it breaks down – as it most surely will – do not throw it away. Hang on to it.

One day it may be a valuable memento of a short period in the history of Australian personal computing.

Gareth Powell

The most important computer in the history of Apple – the Apple IIc

by Gareth Powell

The Apple IIc marks a significant point in the history of a manufacturing company that has changed the face of computing world-wide.

When the Apple II was originally launched it soared away, making personal computers respectable and Wozniak and Jobs millionaires many times over. It brought computing within the reach of millions.

From 1977 until the beginning of this decade, Apple led and the others followed. But, as the years passed, the dream inevitably began to fray at the edges.

Cupertino ceased to be a computer Camelot. The perceived technological lead started to diminish.

Marketing decisions were implemented which in retrospect seemed to have been mistimed. Apple share prices on the United States stock market started to sag.

It was not so much that the Apple was getting oldèr but rather that new and exciting machines were popping up all over the place.

That they were soon to pop down again did not alter the fact that they began to give Apple the image of a company falling behind in the computer race.

The image of the company was not enhanced by the launch of the Apple ///. It had none of the fizz and sparkle we had come to expect from Apple.

Incorrect forecast

There were many who at that time started with a secret glee to forecast the demise of Apple.

Then came the Lisa. The first sign of a new day dawning.

This most remarkable of machines showed that Apple still had the corporate computer genius which could project into the future and then produce a machine which would fit and tailor that future.

True, the Lisa was not for everyone.

True, the pricing policy at the beginning was off the mark.

But the concept, the design, of the machine was so breathtaking in its innovation that the Lisa gave a much needed face-lift to the Apple image.

At the same time, new professional managers had arrived on the scene who could connect technological genius with the real world and make it acceptable.

From this came the Macintosh.

Sales target

Apple announced that if they sold 50,000 of the Macintosh in the first 100 days they would consider they had a success on their hands.

They sold 72,000, and would have sold many more if it had not been for supply problems.

In Australia the Macintosh has yet to completely find its feet – again, simply because of supply problems.

Customers cannot buy machines that are not there.

Shops cannot supply from stocks that are unavailable.

That this will change there is no doubt. Probably well before the end of this year.

The launch of the Macintosh had a secondary effect on the fortunes of Apple in that it gave the Lisa a new lease of life. Retrofitting the Lisa with new drives made the two appear part of a coherent family.

The Macintosh and Lisa together gave Apple a full range of 32 bit machines.

Most importantly, the Macintosh announced to the world that Apple had no intention of bowing down to the might of IBM and that in the world of the personal computer Apple was, at least in the technical sense, the game to follow.

The Macintosh addressed itself to that most modern of diseases, computerphobia.

Apple went out of their way to produce a machine which an absolute beginner could master in hours rather than days.

A machine that could be looked on as a part of a senior executive's desk furniture.

A machine that implied its owner

had computer savvy.

Little wonder it was a tearaway success.

Education market

This 32 bit line of machines was targeted directly at the tertiary education market.

Apple set their sights firmly on that goal and in the United States made it almost totally theirs.

In Australia the jury is still out, but we believe that the Mac/Lisa line will be the dominant force in tertiary education computing for the next ten years.

In the short term we are seeing a large percentage of students in America and Australia working away diligently at their Macintoshes.

In the medium term we are going to see hordes of new graduates coming into the marketplace with their major experience in personal computers being in Apples.

Having control of this area effectively means that Apple has secured its prospects for the foreseeable future.

(That future is also secured in many other ways. Apple has one of the lowest debt ratios of any American company. It has a massive financial reserve which could, on our computation, keep it going for three years without it having to sell one computer. When a company has got to that state it will need war and pestilence and flood and famine to stop its momentum.)

The portable Apple //c, showing easily portable flat display monitor (to be available late in 1984), modem, carrycase and Apple Mouse II.

Having cornered the educational and scientific markets (and several other markets as well) with the technological brilliance of the Macintosh/Lisa line, Apple has now elected to show another area of its talents – marketing knowhow.

Sales sense

There is not much debate that from the beginning Apple needed professional help to maximise the sales of its salesworthy products.

When John Scully arrived from Pepsico to run Apple with a reputation as big as all outdoors, he set up a marketing strategy plan which is now starting to bear results.

We believe that if the Apple II was Act I, Scene 1, then the Macintosh and Lisa brought that first act to a successful close.

Curtain descends to tumultuous applause.

The interval has passed and now the curtain is rising on the second act – Apple launching the IIc and a range of peripherals to complement it.

The IIc is, in our opinion, the most important machine in the history of Apple.

It contains very little new and exciting in the way of technology. Rather it is a synthesis of what the market needs now, not what it may need in some ill-defined future.

When we first saw the machine we said that in our considered belief it would within one year break the sales record for Apple in Australia – computing the figures from the date that Apple first started here, and given the provisio that sufficient numbers of the machine were available.

Early dealer reaction shows we were not wrong.

We do not know what Apple have targeted as their sales figures for the coming fiscal year. If they have set their sights at less than 30,000 Apple IIcs, then we believe they have underestimated the potential of the market.

What is this remarkable machine and why do we think so highly of it?

Read the following story.

The CPU of the Apple //c. The built-in disk drive is on the right-hand side of the unit.

The Apple IIc – review

The standard features of the Apple IIc are as follows.

The CPU is a 65C02 eight bit microprocessor.

This is a logical extension of the standard Apple 6502 CPU but, essentially, uses CMOS technology to allow it to operate with far less power and far lower temperatures.

As it is a lineal descendant of the 6502, almost every program for the Apple II (and there at least 15,000 of them) will run happily on the IIc.

The exceptions are few and far between, but unhappily include Zardax, our favourite word-processing program. No doubt Ian Phillips, the man who is responsible for Zardax, will be offering a new version.

Thanks for the memory

There are 128K bytes of RAM, which is an elegant sufficiency.

It shows how times are changing when no one comments on the size of this memory working with an 8 bit chip.

It is only eight years ago that we were delighted because we put together a machine with a massive 1.5K bytes of RAM. And went round boasting about it.

This size memory means that no matter what program you are using – be it word processing, spreadsheets or databases – there is very little likelihood of you ever running out of memory. If you do, you probably shouldn't have been using an Apple IIc in the first place for the task you have tackled.

However, Powell's Third Law of Computing says: "Programs always expand to fill available memory."

There are programs available on other machines which require up to 200K RAM to work properly. This is not always due to sloppy writing on the part of the programmer. Frequently it is because there are more and more desirable features that can be added.

An example is the word processing program MultiMate, used on the IBM PC. This uses up nearly 200K of RAM when loaded.

You might think this a profligate waste of memory except that the result is a program that turns the computer into the most powerful word processing unit we have ever tested.

So although 128K bytes of RAM seems – and indeed, is – a lot, it may start to look less bountiful in a few years' time.

By then we will have seen the Macintosh memory retro-fitted with 256K RAM chips. And we will almost certainly have entered the era of one megabyte memory chips.

No doubt the Apple IIc will be modified as such memory expansion improvements become available and programs that require these vast amounts of memory become commonplace. That time is some considerable way down the track.

There are 16K bytes of ROM with Applesoft BASIC, the machine language monitor and disassembler built into ROM.

Disk drive

The disk drive is a built-in Shugart drive, and takes what everyone illegally refers to as five and a quarter inch disks.

The logic behind the choice of this disk drive is impeccable. It means that the standard Apple programs can be used without a hiccup.

It is a fair bet that there was much agonising in Apple over this decision.

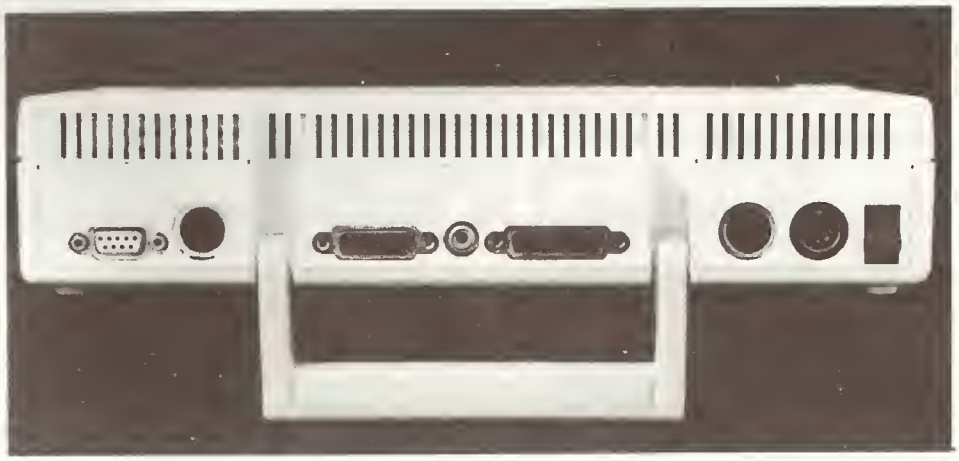
What might be referred to as the 32 bit push would undoubtedly like

to have seen a version of the small drive, which is used in the Macintosh and the Lisa, used in the IIc.

Looking at the IIc carefully and using a metric ruler and a calculating machine, I think it just might marginally have been possible to squeeze two of the small disk Sharp originated drives into the IIc.

But I have no doubt whatosever that the decision made was the right one.

Technically the other disk drives have the edge. But the hassle involved in getting programs sorted out would have negated all the advantages these drives would



have brought.

Fitting this drive appears, to an interested outsider, to have been a pure marketing decision. As such, it is an interesting aspect in the continuing process of the maturing of Apple.

The IIc has, of course, a built in 80 column facility. Those of us who say "about time" are elderly carping critics who should be ignored. The display, which has quite elegant and eminently readable letters, is, of course, in upper and lower case.

The keyboard has 63 letters. Because the Apple IIc is, in intent and in effect, a portable, the keyboard does not have a separate numeric pad, an understandable and acceptable decision. But we hazard the guess it is only a matter of time before some entrepreneur offers it as an add-on, possibly working through the port used for that damned mouse thing.

Keyboard

I seem to spend my life writing. I have used keyboards of all shapes and sizes.

I have wept tears of distress at the design of the IBM keyboard, on which the return key has been moved so that you keep banging the wrong key by mistake.

I have been immeasurably distressed by the Sord portable (which exists) because it has no space bar. Even on the original Apple II I was less than pleased with the keyboard – the IIe was a major advance.

The IIc is better yet. It is almost the perfect keyboard. Indeed, I think it is the state of the art as far as personal computer keyboards are concerned. It has slightly dished keys (rather more dished than those on the IIe) and they have an excellent feel. I sat down at the IIc for the first time and within two minutes was typing in a longish article with no problems whatsoever. It fits into the overall design of the machine, which is a thing of beauty and a joy forever.

There is a built-in loudspeaker, only one channel but it has a volume control, a blessing in these days of noise pollution.

You can fit a joystick or a mouse or any other type of handcontroller.

There is a built-in serial interface for a printer – for which relief, much thanks – and a built-in serial interface for a telephone modem.

Portability

The machine works from 12 volts, at the moment supplied through a transformer. There is no doubt that this machine will work extremely well on batteries and, indeed, on at least one occasion in Australia it has already done so, using the battery pack from a video recorder unit.

We understand that towards the end of this year the IIc will be available with a battery pack, an LCD screen and a keyboard which can be changed over from QWERTY layout to the infinitely preferable Dvorak.

The story has been going around for some time that Apple was going to make a big technological break through in flat screens, and the fact that they have opted for the something less than perfect LCD screen has caused some raised eyebrows.

In fact, it fits in well with the whole philosophy behind the IIc. It is not on the cutting edge of computer science – that is where the Lisa and the Macintosh have staked their claims. Instead it uses modern but tried and true technology to bring the mostest machine to the public for the lowest price.

When the machine with the flat screen becomes available before the end of this year, we will have a true Apple portable which, in truth, is going to destroy the market for a large number of other machines.

There will still be room for the smaller lap machines such as the TRS80 100 and the NEC, but the portable market – computers which are fully self contained and portable but are still full computers in their own right – will, I believe, be almost totally dominated by the IIc.

In my case, as an itinerant journalist and publisher who concentrates on computers and travel, the machine is the answer to all my prayers.

Wherever I go I can take a fully-equipped office with me. All built into an Apple IIc.

The design of the IIc comes from a German company called Frog (strange name that) which was responsible for the design of the Sony Walkman. (One day I must write the story of how the concept of the Sony Walkman came about and why one airline has learned the sad lesson that you must always keep your big mouth shut when you have a good idea.)

It is a magnificent piece of design and is, in truth, a timeless work of art.

Simple features delight.

Take three factors.

The machine needs air circulation under the base to keep it cool.

The keyboard needs to be angled for the best ergonomic positon.

A portable needs a carrying handle.

In the Apple IIc all these needs are catered for with one neat design feature.

The handle props up the case to the correct angle, allowing air circulation. And at the same time it makes the IIc as easy to carry as a medium weight brief case.

Documentation

The Apple IIc comes with the most comprehensive set of documentation and learning programs that have ever been seen for a portable computer. They have set a standard which will be difficult for other manufacturers to attain.

They are, in a word, superb.

If you read the books and follow the disk introductions you will know all you will ever need to know about your machine. And the process will have been painless.

For the fringe of computer maniacs like myself and for programmers, there is the Apple IIc Reference Manual which tells you more about the IIc than is decent for a young lad to know.

To complement the Apple IIc there is a new printer, the Scribe, referred to elsewhere in this issue. There is also a monochrome monitor.

The design of all these peripherals is of the same standard as the IIc, and I have absolutely no doubt that the whole assembly will be added to the Museum of Modern Art's permanent collection of superbly designed machinery on display in New York.

To sum up, if anyone had any doubts about Apple's ability to turn out a magnificent machine for all reasons and all seasons, those doubts have now been set permanently to rest.

When you can't afford mistakes.

There really is a difference in diskette brands.

It's in the way they are made.

Because the computer industry cannot afford variation in the quality of diskettes, at Nashua we looked for, and found a way to ensure absolute consistency.

Here's how we do it.

Quality Circles.

At Nashua we've found the best way to attain this 'consistency' is to ensure that at each stage of production our diskettes

are statistically checked to make sure the quality is 'built-in' every step of the way.

Rather than long production lines, we have 'Quality Circles' – small groups of people whose job it is to make sure that each Nashua diskette is right in the first place.

The result is a diskette with such consistency, that it is chosen by those people who can't afford mistakes.

Phone Sydney 958 2044, Melbourne 428 0501, Brisbane 369 4244, Adelaide 42 0021, Perth 328 1888, Hobart 34 3761, Darwin 81 6204.

Nashua

There really is a difference.

Ruwald & Skinner 400NA

AT LAST!

A SENSIBLE ALTERNATIVE TO THOSE FAR-AWAY MAIL ORDER SUPPLIERS! CALL INTO OUR COMPUTER CENTRE FOR A DEMONSTRATION BEFORE YOU BUY. ALL ADVERTISED ITEMS NOW IN STOCK!

EAGLE ELECTRONICS INTRODUCES

Super 5®

THE BEST QUALITY FLOPPY DISK DRIVE FOR YOUR APPLE® IS THE *Super 5*

SUPER 5 SYSTEM (T-40)
(Red Label)

- Direct Drive TEAC mechanism.
- Metal Band positioner.
- 13 or 16 sector.
- 40 tracks – density 48 TPI.
- Track to track – less than 6ms.

only $379

NEW TYPE

LOW POWER CONSUMPTION THE FINEST QUALITY

CP-80 PRINTER

Ask us for a demo of this very popular printer – you'll be surprised by its advanced features and low cost. Interfaces to all popular computers like APPLE and IBM with Centronics interface – RS232 interface also available.

only $399

MULTIFUNCTION BOARD

For IBM-PC or compatible computer

only $549

including 128K RAM!

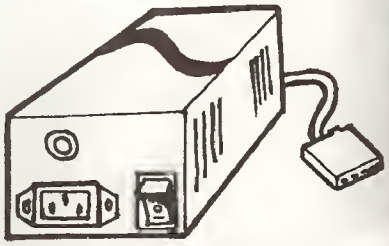
- 64K-256K RAM
- Centronics printer interface
- RS232 Port (2nd RS232 optional)
- Real time clock with battery backup
- As used in CONDOR computer

APPLE POWER!

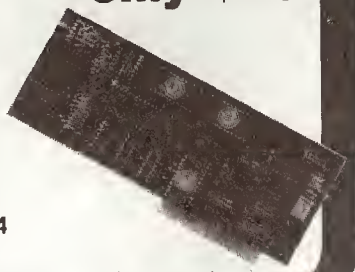
– has your Apple power supply given up?

THIS ONE won't!

* Thumping 60 watt rating.
* Fuse protected.
* No cooling fan required.



only $98



ALL PRICES INCLUDE SALES TAX

ACCESSORIES for HAWK and APPLE

Disc Controller	$59
80 Column Card	$99
Z80 (CP/M) Card	$69
Epson Printer Controller (with cable)	$79
RS-232 Card	$74
EPROM Burner	$95
PAL Card + modulator	$104
Joystick	$27

® Apple is a registered trademark of Apple computer inc.

Eagle Electronics Computer Centre
83 Princes Hwy., St.Peters. (cnr. Short St.)

bankcard

Eagle

P.O. BOX 161, ST. PETERS, N.S.W. 2044, AUSTRALIA. TELEPHONE: (02) 516 4540

Errors and omissions excepted

Apple make Samurai agreement

Hands up those who thought they would never see it happen. In the past, Apple have shown signs of advanced xenophobia.

No more. Now they seem to be well and firmly entrenched in the land of the geisha house and "zaibatsu". Latest news is that Apple have got into bed with Sony Corporation to share existing research on hard disk technology. To quote from the information given by Apple: "Apple will provide Sony with its existing research on hard disk technology, enabling Sony to expand its product line of computer peripheral devices." Nothing is said about what Apple are getting out of the deal, but you can make the presumption it will be a similar amount of research information.

Our guess is the result will be hard disk drives in the 3.5 inch range (as they are still called) which will fit exactly where the Sony 3.5 inch drives fit in the Lisa and the Macintosh, but these drives will give memories of several megabytes each – probably four.

It is interesting to see that industry newsletters in the United States have stopped using the phrase megabytes and are now on to gallons. So our guess is that the first fruits of this cooperation will be a four gallon job on a 3.5 inch hard disk. Could even be five gallons.

Hitachi for Apple

For owners of Apple IIes the following announcement, though short, is only slightly short of momentous.

Hitachi has introduced a 3 inch (72 mm for the enlightened) disk drive for the Apple IIe at about $300 a pop and you can use them with a standard drive or you can use two of them as drives one and two.

Maxell have produced the disk to go with it. This disk has, and here we must give you the advertising copy in all its full, florid beauty, "epitaxial magnetic material, the first of its kind to be used for consumer products, consists of cobalt ferrite encapsulating the gamma ferric oxide". And we always thought it was the same sort of stuff they put in Vegemite.

This exotic blend allows you to put about 286K, unformatted, on a disk, which as Damon Runyon is wont to say is more than somewhat. The suggestion exists that in the near future this will be increased to over 500K.

And why do we write about these wonders in an Apple magazine?

Because this device hangs on to an Apple IIe with the minimum of fuss is why.

It fits right on to your present disk controller card as to the manner born. If these become readily available at the cost of $300 each – and that is the price we have heard suggested – then we are going to think very carefully about switching to the new technology. Because in our opinion these smaller drives will inevitably be the way of the future.

It is possible, if not probable, that the Apple IIe will move in this direction over the next year or so. If that were to happen then the IIc could possibly be supplied with two built-in disk drives, which would make it the best personal computer in the world.

For Apple IIe users this short paragraph is momentous news indeed. Remember you read it here first.

Hitachi Compact Floppy Disk

Classroom magazine

Ashton Scholastic is the Australian subsidiary of the American publishing company Ashton Scholastic. They now have a new magazine called "Classroom Computing", intended for the use and benefit of teachers. Many of the programs and books discussed are either produced by Ashton Scholastic or distributed by them. This does not in anyway lessen the worth of the magazine, which is well written and designed.

If you are of a cynical turn of mind you can catch, when leafing through the pages, a faint echo of the desperation which many teachers feel when faced with teaching a class on a computer. Because in most cases members of the class know far more about computing than the teacher does. It would be like teaching German to a lot of German migrants. Not a happy thought. The magazine takes infinite pains to reassure the teacher every step of the way. In many cases where it is showing the sort of information that can be taught to students, there is no doubt that it is subtly teaching the teachers first.

In a Sydney school which we know quite well, a youth managed to manipulate the school's computer so that the end of term tests set by the headmaster were locked with an encryption code, and the combined efforts of the teachers and the headmaster could not crack the code. □

People who hear stories like this shake their heads in rueful admiration and say how smart the children are becoming. The headmaster was made of sterner stuff. He caught the culprit and set him a small task – writing a program in BASIC to forecast population growth within selected areas of Australia. Each time it is finished the headmaster finds at least two bugs and sends it back for rewriting. As we go to press the lad is moving in to his ten thousandth line of programming code.

That will keep the little ankle biter in line.

Apple publish for education

A new magazine called "Apple Australia Education News" is being published by Apple Australia. It is edited by a lady called Diana Ryall, and is more of an eight page newsletter than a magazine.

At a quick glance, this magazine seems to contain nothing but information on Apple products which, it is true, is of great interest and benefit to teachers. But articles about the place of the computer in education would not come amiss. They could still be Apple orientated but be of more direct practical value.

It is true that the Apple is probably the best personal computer for schools, and it is also true that a magazine showing how to make a personal computer work in schools would be a good thing.

Ashton Scholastic with "Classroom Computing" appear to have caught the hang of the idea. Apple with its "Apple Australia Education News" has some way yet to go. As only a few issues have been produced, it is early days yet. We will watch progress with great interest. If you are a schoolteacher, your school gets a copy free.

Wider still and wider

Apple have announced a new printer that will take paper up to 38 centimetres wide. It is called an Imagewriter and uses a 7x9 dot matrix (which gives damn near letter quality printing) at up to 120 characters per second. It is, of course, a wide bodied version of the standard Imagewriter. We were never overly impressed by this machine, considering it but a clone of the C.Itoh 8510 which we already use to our total satisfaction. A bit noisy but relentlessly efficient and totally dependable.

Our eyes were opened by Apple Australia's Mike Bolan, who showed us what the machine could do in the way of type and graphics. It was a revelation to us. The Imagewriter in this size is perfect for producing spreadsheets and accounts. Price is $1,195, which makes it probably the cheapest printer of its size available in Australia.

With it you can produce a set of accounts which will convince your bank manager about your aesthetic qualities, if not your financial capability.

It is a better bet than buying the equivalent C.Itoh direct from Warburton Franki, the Australian distributors. The prices are just about the same, but the Imagewriter works far better as an Apple peripheral.

Elsewhere in this issue we review Apple's new printer, just released for the IIc, which prints in colour. Not yet available with a wide carriage but definitely a sign of things to come.

In our next issue we will be reviewing a Hewlett Packard ink jet printer specially configured for the Apple, which from first appearances seems set to become a major best seller.

Awesome Ausom

The Apple Users' Society of Melbourne (membership enquiries to Grahame Willis, home number 03 878 0219 or David Halprin, business number 03 387 3221) produces a lively magazine called Ausom News. In their magazine library (run by Peter Eanins, home number 589 2072) they keep copies of Australian Apple Review, which may not help us financially but is very good for the ego. They run a pretty lively magazine, and we have asked one of their writers to provide an occasional article for us. As well they have a sub-group for the MedFly – the machine we keep forgetting to review. This sub-group is called MEDUSA – which, as does AUSOM, sounds like a terrorist organisation in a James Bond movie – and reading their report one can see hints that the major problem with the Medfly is documentation and manufacturer support. As we know much of the history of MedFly we are not surprised.

Sleekit beastie

We have an awful feeling we are losing our mouse extermination campaign. Now it is available on the IIe and according to Apple "will be offered as an option with most new software packages from Apple for the Apple II series computer".

Apple IIe, DuoDisk drive, mouse and monitor showing MousePaint graphics

In the picture is an Apple IIe system with an Apple DuoDisk – an add-on about which Apple Australia appeared to have been strangely reticent. In fact, it was launched one week before the Macintosh, with the result that its announcement was overshadowed by the appearance of the mighty midget.

On the screen is an exploded view of the mouse (we'll drink to that) and the worktool menu for the MousePaint graphics package. Not a million miles away from the MacPaint package on the Macintosh.

Readers who are also amateur photographers are asked to consider the clarity of the screen. Bearing in mind this is a green phosphor screen, how did they get such clarity of detail in the picture? If you light the computer enough to bring out the detail then you must get reflections on the screen even if you are using a tent or a spray to kill reflections.

We are almost certain this shot was taken in two stages. In the first stage, a piece of black velvet was cut to the exact shape of the screen and held in position while an exposure was made to show the machine.

Then the velvet mask was removed, the lighting illuminating the machine switched off and the second part of a double exposure made to reveal the details on the screen. There may be other ways of doing it. Any suggestions?

Apple news

At the University of Wollongong, NSW, there are now 30 Macintosh computer work stations in operation. This is the first major Australian installation of the new machines. The new laboratory has, informally, been called "Skylab".

It may be considered strange that Wollongong University should have the first major batch of Macintoshes, but that university has always been at the forefront of matters scientific in Australia.

For example, in 1976 the computer science department was the first in Australia to introduce regular, supervised laboratory classes in computers for first year students. In 1977 they were the first in the world to make the Unix operating system transportable between computers.

We are not quite sure what good this did for the world at large, and we would never accept the university's assertion that the Unix operating system has significantly reduced the complexity of using sophisticated computer systems. That is, in our unhumble opinion, a load of old cobblers. If anything, Unix complicates matters.

However, Wollongong is the first cab off the rank with the Macintosh. It will be interesting to see how many Australian Universities follow their lead.

For what it is worth our guess is that the major sale of Macintoshes will be into universities and we think within the year we will be able to announce that every university in Australia is equipped with Macintoshes. □

Computer languages – a study in comparative religions

by Duncan McCann

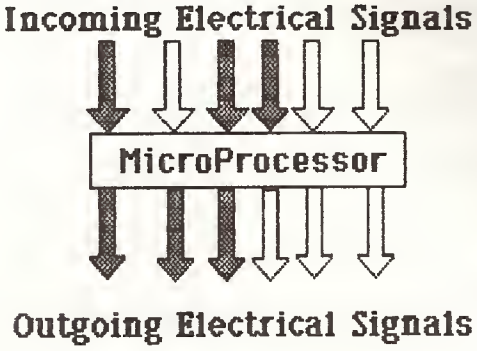
It has been held that computer languages are like religions. Discussion on the subject, whilst starting out rationally, rapidly turns into a series of accusative assertions and emotive statements. A person's chosen language often reflects deep aspects of their personality and a vision of the hereafter – or rather what sorts of rewards we think are deserved for what sorts of effort and privation. There are punitive languages demanding long and painful devotion for little reward and there are pie-in-the-sky languages promising wonderful rewards for little effort.

Why is this and why are there any languages except BASIC anyway? Each different computer language has its own particular strengths and also weaknesses. A language for music synthesisers probably won't be too useful in controlling the space shuttle, so there are many languages available to us. To explore this matter, all we need to know is that a language allows us to write a list of instructions or program.

Let's explore further . . .

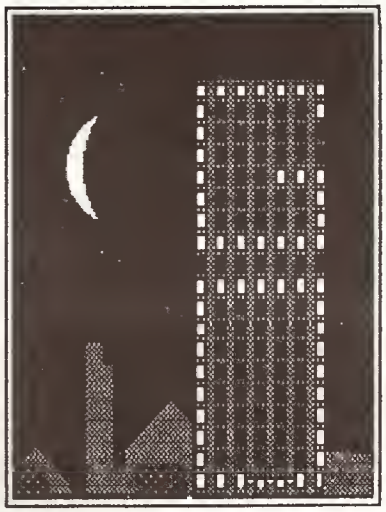
Machine language (binary)

The computer itself doesn't respond to anything except a series of electrical impulses. These impulses can be represented as 0 or 1, either there's an impulse or there isn't.



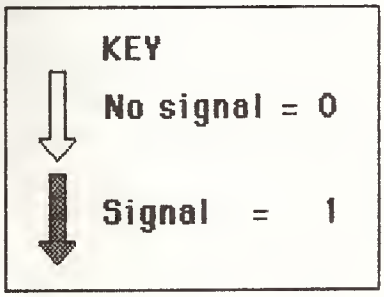
The Microprocessor modifies signals it receives according to how it is micro programmed by the manufacturer.

OK. Now let's take a look at how we can use this information. Suppose you wanted to send a message at night to a passing aircraft but you couldn't use radio. You could arrange to switch on lights in some tall building and have the pattern created form a pre-arranged message:

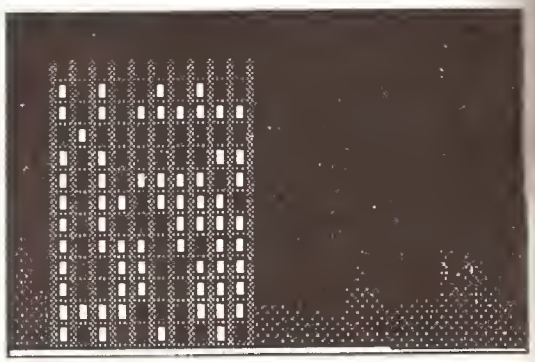


Although controlled only with on/off light switches, complex messages can be built with patterns.

Computers use this method of saving and using information except instead of light switches they use microscopic circuits which either contain an electric charge or not and are interpreted as 1 or 0.



If we had paid the caretaker of a building to set the lights for us, we would have had to give him a **list** of how to set the lights for each floor. This list is the same as our computer program, which is also a list of binary machine code instructions. Naturally you'd have to take care that you only turned on the appropriate lights; if you made a mistake then your message would be different from the one you intended. If this occurred in a computer program it would be called a "bug".



Complex messages can easily be coded using other patterns.

Checking that the caretaker hadn't made any mistakes in a complex message would be a tiresome process indeed. Machine language programs are hard to "de-bug" for the same reason.

Assembly language

The list we'd have given the caretaker would show the light switch settings for each floor. Somewhere, however, we'd need to have another list which told us what each of the patterns stood for, so that we could set the switches accordingly. This list would have the word corresponding

to each setting and would enable us to compose our messages more easily than trying to think it through in switch settings.

An **assembler** is a translation program that converts our commands into switch settings (or binary), thereby saving us a lot of trouble and reducing the likelihood of mistakes. The language is known as **assembly language**. It is fast, efficient and usually takes up a relatively small amount of space in our computer but it is also complex, tricky to learn and hard to understand once written. Assemblers let us write instructions like JUMP instead of 01001100 because the assembler program translates JUMP into binary code.

There are relatively few instructions that a computer can carry out, perhaps 40 to 80 depending on the type of computer. This is not enough to calculate things like % or the sine of an angle or even to multiply. To do this kind of thing programmers use the instructions provided with the computer in groups or lists and call the sequence of instructions whenever they need them. A typical multiplication routine can take 10 to 50 lines of code in assembler.

FORTH

Forth is a language based on Assembler which allows the programmer to group instructions any way she wants. She may want a group which draws a square on the screen, for instance. With FORTH she can build up such a group and identify it with a unique name such as "square". Later when she uses the word "square" in a program the computer automatically carries out the instructions in the list. This would also be a handy facility for our caretaker setting the lights if he often had to repeat switch settings for different floors; certainly it would save us a lot of writing if we directed him to set floors 18 to 23 the same as floors 7 to 12 for instance. Knitting patterns work the same way, as do other lists of instructions such as those used in cooking.

With this technique the computer language can be extended to our own particular needs, and computer languages which have this facility are known as "extensible". Extensibility saves the programmer much work later on and greatly increases her ability to solve problems. Each new command can become part of the FORTH language and the user can build up different versions for different problems, eg a word processing language and a graphics language. In this way FORTH can become a very "personal" language and very efficient for the individual. Some high level commands are already part of the language, multiply is just one instruction for instance.

C

C is a more structured language which also gives the programmer access to many of the features of assembly language without its great detail. Because of the design of the language it is usually relatively easy to run C programs on different computers, a process called "porting". C is mainly available to users of the UNIX operating system found on large computers, but is just becoming available on microcomputers.

BASIC

A "shirt-sleeves" language which allows a beginner to do rough and ready problem solving. BASIC also has relatively few commands and, unlike FORTH, is not extensible. BASIC comes in two major flavours:

Interpreted – each instruction is carried out by the computer one line at a time, just like a pocket calculator, OR

Compiled – first you type in the whole list of instructions, then the computer runs through them all at once.

The compiled method is faster, but you don't find out about mistakes until you've typed everything in. An interpreted BASIC can tell you if you've made an error each time you enter a line. Since beginners, for whom BASIC was designed, are prone to making errors, most people prefer interpreted BASIC, at least to start with.

Because of the limited number of commands, BASIC can get very complicated when writing long programs. Everything tends to look the same and understanding the program after it's written can be tough. New versions of BASIC are being written which offer "advanced features" contained in other languages. Sort of like sawing the top off your Holden wagon because you want a sports car or building a camper on your Ferrari!

Pascal

An extensible language designed to help people learn better programming. Pascal features English like commands and programs which are easily understood. Pascal is becoming very popular with the availability of larger memories for micros. It is also a language of choice for many business applications. Major drawback is that only compiled versions are available, although Macintosh will have an interpreted version available soon (see BASIC types above).

Fortran

A language designed mainly for scientists and engineers and hence popular in University environments. Some micro implementations exist but have a limited market. It's the language of choice for solving scientific equations.

Lisp

Lisp is a complex language that produces programs which can modify themselves. This "symbol manipulation" language is very

popular with Artificial Intelligence researchers and simulation programmers.

Logo

A language marketed for educating children, but useful for any age level as it also has many elements of Lisp. Logo features ease of use (children age 6 can use Logo) and stresses learning by discovery and graphics. Logo is extensible and very powerful although the system is usually big and relatively slow. Logo is extremely popular in schools, particularly primary, for teaching maths, physics and problem solving skills.

SmallTalk

SmallTalk is an innovative language designed for ease of use, graphic output and relatively easy programming of complex phenomena, and is therefore most useful for complex simulations. Most beginners don't need to concern themselves with this type of language as it is very hard to obtain. Many of the features of the Mac/Lisa product range are based on SmallTalk principles.

What to choose

We've surveyed the major languages from lowest to highest level, ie those having instructions most like machine language (switch settings) to those with very English like structures.

What to buy? If you are a beginner unsure of your abilities and you'd like a powerful but easy to use language you should consider Logo. Perhaps as you gain more skill you may want to switch to a compiled language like Pascal.

Remember that writing long programs is a complex and time-consuming job, and you may want to buy a program which has been designed to solve your particular problem without learning a computer language at all! □

An approach to Fortran

by **Tony Cane**

I learnt to program with BASIC.

The programs I had been writing were to analyse information gathered from instruments, and I felt that BASIC did the job well. In any case at the time I wasn't aware that any other programming language would be superior at the tasks involved.

However, recently in a new job, I began to use FORTRAN. While initially FORTRAN and BASIC looked as though they were dialects of the same language, noticeable differences emerged when I began constructing my second program.

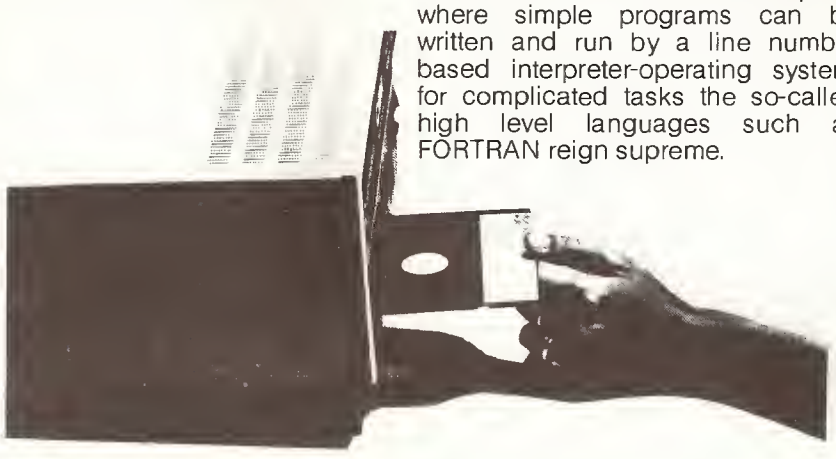
When you write a program using BASIC, unique numbers are given to each line of the instruction and unique names to each variable. This makes BASIC easy to use and debug because you can follow the values of specific variables while tracing the path the computer is taking by having it print line numbers as it works.

However, as programs begin to get complicated, this advantage of BASIC begins to lose its value. Instead it is easier to assemble together sections of code each of which does a specific job, such as sorting a list of names or plotting a graph, that have already been tested and debugged. Such mini programs are called subroutines or procedures.

But when this is done using BASIC it means that all code must be given line numbers that allow the block to function within the program, and variable names that match those in the main program. Worse yet, if the same operaton is to be done to many different variables, either separate blocks of code must be used or complicated transfer subprograms written. And even worse still, if you have a useful subprogram or function it must be rewritten every time you want to use it in different programs to ensure that it has the correct names and suitable line numbers.

In contrast, with FORTRAN, subprograms are written as blocks of code to do a specific job, and the whole block is given a name. When that operation is required, it is called to do its function by this specific name. To overcome the problem of mismatching variable names, one part of the subprogram name is a list of the variables required from the main part of the program. When the subprogram is called, this list of variables is filled in with the names used in the main section of the program, and when the subprogram matches this list with the values it uses, away it goes. This means that after the code for an operation has been written and checked to be sure it works, it will never need to be rewritten; instead it can be put into a library till needed.

So while BASIC is useful in the environment of a microcomputer where simple programs can be written and run by a line number based interpreter-operating system, for complicated tasks the so-called high level languages such as FORTRAN reign supreme. □



Writing machine language

by Gareth Powell

Hands up all those who have a tremendous feeling of inadequacy when they read that a program should be written in assembly language or machine language.

It would be the vast majority of readers of this magazine, including the publisher and editor.

We all know that snazzy arcade games are written that way. As are many serious business applications. And yet we trudge along using BASIC and swearing at its slowness and limitations.

It is my personal theory that writing or assembling a whole program in assembly or machine language is not the way to go.

With BASIC you can write a program using lots of REMS – those splendidly named remark lines which do nothing to the program except remind the programmer what he was doing when he wrote a short cut that would speed up the program by a factor of five and ended up going in to a perpetual loop with the only escape route the Control and Escape buttons.

With BASIC you can, using those REMS, get all the major bugs out at an early stage.

With BASIC you can easily modify a program after you have run it for a while to incorporate improvements of an operational or stylistic kind. Show me a programmer who is totally satisfied with a program after he has finished and I will show you a programmer with no soul.

Why then do we need machine language?

Speed, speed and more speed.

Writing relevant parts of the program in machine or assembly language when speed is of the essence, and the rest in BASIC or any other High Level Language, appears to my simplistic mind to be the way to go.

When you need speed you have it – between ten and a hundred times faster than BASIC depending on the routine – but the rest of the time you have debuggable clarity.

Anyone who tells you that programming in machine language is easy is talking through the top of their pointy head.

On the other hand it is not beyond the average intelligence either.

Regard carefully any machine language programmer and you will not be impressed by a blazing intelligence.

Programming in machine language requires a methodical approach and a willingness to learn systematically. Knowledge does not in this case come as a burst of blinding light. It is a steady acquiring of understanding.

With the Apple the only way to learn is by hands-on experience.

Doing is learning.

Most of the books on machine language are obscure and obtuse to a degree.

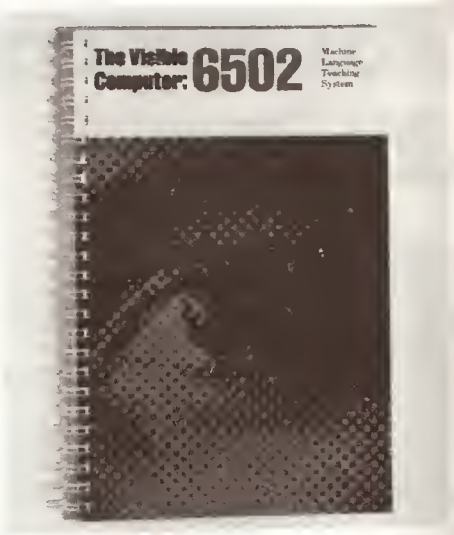
Far better is a program called "The Visible Computer" by Charles Anderson, which was awarded the title of "Best Microcomputer Software of the Year, 1983" by a group of computer education journals in the United States.

It is an excellent program. It consists of a 142 page spiral bound paperback and a floppy disk.

The Visible Computer starts off on the premise that you understand at least the fundamentals of BASIC. Then it takes you firmly by the hand and guides you through the intricacies of machine programming for the Apple.

The book starts by going through the standard material on computer architecture – the way the memory is stuck together – and number systems.

After you have finished the first few chapters you will understand quite clearly the difference between binary and hexadecimal and why the Roman Empire could never have invented computers (have you tried dividing LCXXXIII by CCXXII lately?)



After that the book gets stuck into teaching machine language in a most readable way, and uses the program to emphasise and underline and explain the difficult steps.

This is one of those rare packages where the program and the book work together and are written as a cohesive whole. The program uses the Hi-Res graphics screen to display a diagram of the 6502 processor with its internal registers. It brilliantly simulates the action of the processor, showing all the internal changes that occur during execution.

It doesn't just give you an overall, broad, simplistic view. It gets right down to the nitty gritty of showing you the individual steps within the execution of a single instruction.

It is all done with a sense of lightness that is pretty to watch. To show data transfer to and from memory you have jolly little animated hexadecimal digits nipping backwards and forwards across the screen.

The program allows you to dissect a machine language program to any level. As you progress through the book you load sample programs into your machine and then work your way through the examples.

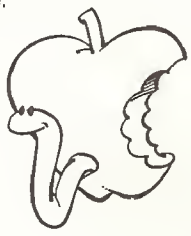
If you buy this package and logically and carefully go through the book and through the examples, you will almost certainly end up being able to handle machine language.

Not perfectly, not with total ease and facility. But adequately. □

There is certainly less than a clear agreement throughout the computer industry as to the meaning of the word standard. This Worm was present at a meeting when a major car company made loud noises about having broken car records which, as far as this worm could ascertain, had not even been set. How you break records that do not exist bewilders me.

The timing of these records, if that is the word I am looking for, was left in the hands of a gentleman using a computer that shall be nameless, a Hewlett Packard. This worm, who has a more than passing interest in matters of this sort, inspected the machine closely and saw it had two drives for the small floppy disks which are encased in a rigid plastic housing. (And, yes, I agree. In that case they are no longer floppy). A charming gentleman of saturnine appearance – caused no doubt by an over-indulgence in yachting – came over and told me these were three inch disks and that they were the new industry standard.

I am never one to argue with a man who is obviously both my elder and better. But have the other manufacturers been told? Are they still manufacturing disk drives which are quarter of an inch or half an inch bigger, in complete ignorance that a new industry standard has been set? Why, if Hewlett Packard know there is a new standard, why do they not tell everyone? There is a lot of expensive machinery out there that has to be scrapped so we can all toe the line.



Rumours have been circulating that the Apple /// is moving towards the end of its useful life. A fellow Worm in Cupertino – that heavenly land across the sea from which all good and truth emanates – tells me Apple management have moved even further on their position that Apple /// is now an independent cost centre, and must make a profit or else.

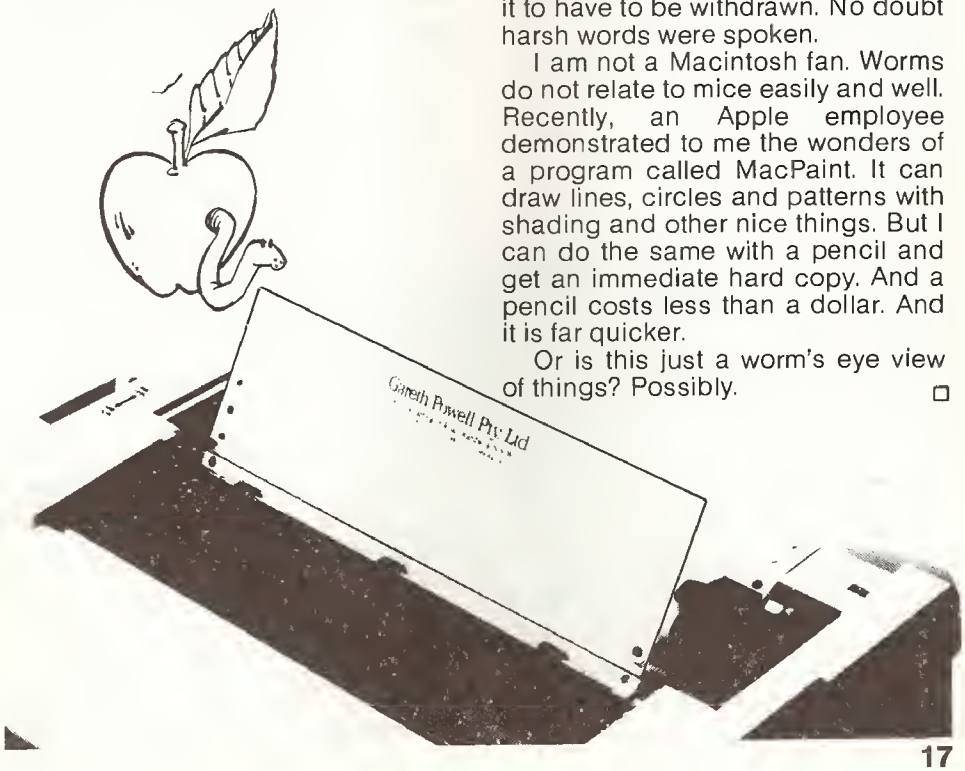
I am told they have decided that not only will it never become a major profit centre (no argument about that one), but that it is affecting the Apple image in a detrimental way.

Being a Worm and therefore of a low level of intelligence, I have the greatest difficulty in following that argument.

True the Apple /// had a launching which could be called less than felicitous. True the early models had enough bugs in to send a computer programmer into a frenzy. But now the machine has been sorted out, it is ideal for use in vertical applications in small businesses such as service industries. This is a role that this Worm can hardly see the Lisa fulfilling – it is too sophisticated a machine for, say, a garage. And yet the IIe may not be quite powerful enough.

I am willing to accept that image is everything nowadays. But I feel Apple would be making a premature mistake if they shot the Apple /// down in flames. After all, there are well over a hundred thousand users out there. What will happen to those poor benighted devils?

Decisions to drop machines are often made for the wrong reasons. I hope Apple have very carefully thought this one all the way through.



The new IIc has been greeted with raptures by all and sundry – sundry being the prematurely aging editor and publisher of this august magazine.

As always, it is left to a Worm to add a querulous note to the proceedings.

Why didn't Apple produce the IIc three years ago?

None of the technology is that new. None of the concepts are earth-shattering. There is nothing basically that could not have been marketed a considerable time ago.

If they are that clever why did it take them so long?

And all I hear is silence.

This Worm has heard Bill Gates speak, and indeed he is Jack-the-lad when it comes to computers and programming. And he is still in his twenties. But he must have been blushing furiously when he found out that MultiPlan, which his company MicroSoft produced for the Apple, had enough bugs in it for it to have to be withdrawn. No doubt harsh words were spoken.

I am not a Macintosh fan. Worms do not relate to mice easily and well. Recently, an Apple employee demonstrated to me the wonders of a program called MacPaint. It can draw lines, circles and patterns with shading and other nice things. But I can do the same with a pencil and get an immediate hard copy. And a pencil costs less than a dollar. And it is far quicker.

Or is this just a worm's eye view of things? Possibly. □

K or k

Dear Sir,

Probably by now you have been made aware of the mistakes which occur on page 8 of your March issue, but just in case you haven't, the following may be of interest. 230 400 000 is equivalent to 230 400k, while 230.4k is equivalent to 2,304,000.

Note also that in the SI system which this country has officially been using for several years, there are no commas used to separate the thousands columns, and the suffix used to designate one thousand units is 'k', not 'K' (Kelvin, a unit of temperature).

Congratulations on producing such a fine publication, and please find enclosed my subscription for the next twelve months.

Ken Batt,
Wagga Wagga, NSW.

Ed: *The sort of letter that makes you glad that you are a publisher.*

Taking the three points separately.

Accepting for the moment that "k" exists, then "k" equals 1,024. And 230.4k is 230.4 x by 1,024 which is not, according to our calculator, 2,304,000.

It is true that when we print thousands and millions we should use spaces instead of commas.

We don't and we are in the wrong.

What was 1,000,000 should now become 1 000 000.

We would like some reader reaction before we make the change. We know the standard exists because we printed a book in Hong Kong following this convention, and we had a hell of a job explaining to irate readers who didn't know what the hell we were talking about.

The key to the acceptance of a new standard is that everyone understands from the word go what is meant.

With decimal coinage we have been assured that $12 million was spent in educating the public.

The same effort went into metrication (but note that computer manufacturers still say 3 inch disk drive in blatant defiance of the law).

We asked News Limited what standards they used on their newspapers. They have someone whose only job is to monitor style to see that such rules are closely followed. No spaces for them, still commas.

Because they want their readers to understand instantly and unambiguously what is being said. And readers understand 1,000 as a thousand much more easily than 1 000. But we would be interested to hear a case for the other side.

Finally, it is true that K stands for Kelvin, but k is not, as far as we can ascertain, officially accepted by anyone as meaning either 1,000 or 1 000 or 1,024 or 1 024. It is merely an useful device which has grown with computing. And it is used normally as a capital K.

We have been ruminating on these points for some time. Ken Batt's letter brought our thoughts into focus.

Likes and dislikes

Dear Ed,

Your specialist magazine is much appreciated. No longer must I munch through acorns, peaches, peanuts and apricots to find my favourite fruit (I own a //e with a PX-80 printer).

I'm sure you like to hear your reader's preferences, so here's my tuppence worth.

LIKES: (in descending order) Assembler and Pascal articles, listings, technical information, utilities, word processing, adventures, hardware reviews, communications, industry comment (hang in there, worm).

DISLIKES: Spreadsheets, Applesoft, Logo, QWERTY keyboards (should be first on the list), the pirating mentality, Albert Langer, IBM, crummy lookalikes, CP/M.

A special commendation to AAR for its matt pages – I notice YC has recently followed suit. A pox on glossy paper in magazines!

Some observations on the March issue:

Numeric keypad – great idea, no listing!!?!!

Communications on the Apple II – Philipson promises to look at some packages, but somehow doesn't get around to it.

Maybe rectify in next issue?

Martin Doherty
Hurstville, NSW.

Ed: *Martin Doherty places his finger on several errors of omission and commission.*

The Numeric Keypad program by Bevan Ting was mislaid by the publisher's wife, who is responsible for editing all copy.
Many apologies. We got another copy from the author, and it is reproduced below.

The communications story is a long and complex one.

We thought we would do it once over lightly. As we dug into it we started to become convinced that communications is the way of the future for personal computers. We tried the "Australian Beginning" and were less than impressed. And many of the communication packages we have seen leave something to be desired.

So we have gone flat out trying to assemble a definitive study of communications and, at the same time, do something practical about it. More on this including the wonders of microwave communication with your Apple, in our next issue.

We're with you on some of your dislikes – C/PM, QWERTY, crummy lookalikes.

But we are quite fond of Logo, and spreadsheets have sold more Apple computers than any other type of program. Indeed, if it wasn't for VisiCalc it is unlikely that Apple would ever have made it in a big way. □

Numeric Keypad

```
ASM


*
]CALL-151

*300.37F

0300- A9 1B 85 38 A0 FD 84 39
0308- 4C EA 03 A0 01 84 06 88
0310- 84 05 A9 1D 85 38 A0 03
0318- 84 39 4C EA 03 20 1B FD
0320- C9 8D F0 23 C9 93 D0 11
0328- A4 06 C0 01 B0 02 A0 02
0330- 88 84 06 A4 05 C8 84 05
0338- 60 A4 05 F0 06 CA 88 D0
0340- FC 84 05 A4 06 F0 05 A0
0348- 01 84 06 60 C9 CA 90 04
0350- C9 CD 90 28 C9 D5 F0 15
0358- C9 C9 F0 14 C9 CD F0 13
0360- C9 BB F0 12 C9 BA F0 11
0368- C9 CF F0 10 60 A9 B4 60
0370- A9 B5 60 A9 B0 60 A9 AB
0378- 60 A9 AA 60 38 E9 19 60
*0


]PR#0
```

The changing world of printers

by Gareth Powell

The whole world around us is changing so fast I sometimes feel Alvin Toffler wrote "Future Shock" with me in mind. Take the simple world of printers. Every time I finish an article on the state of printers for microcomputers, the whole scene changes and I find what I have written is out of date. This article applies only to this issue. By the next issue there will have been major changes. No one has told me there will be major changes. I just know from the experience of the past few months that there will be.

An Apple without a printer is only half a computer. You need a printer to list programs, write letters, produce fancy drawings, compile a mailing list. No matter how poor the quality of the print or the slowness of the printer, it is almost essential.

It is an interesting philosophical thought that the computer does not save paper, does not banish the written word. Instead it uses paper in a profligate manner. An office with a computer will go through far more paper than an office without. Pundits promise us that we are heading towards the paperless office. We will see the day of the paperless office on the same day that we see the advent of the paperless toilet.

Bewildering choice

If you intend to use your Apple in business, eventually and inevitably you are going to need a printer. The choice bewildering.

In our continuous search for simplification, here is a quick run down on what is currently available this month – and some indications of which way printers will go, which means what we expect to see next month.

First came thermal printers. I remember using my Apple Silentwriter with joy and pleasure. It was not easy to load the paper and the type faded after six months. But it produced listings to my great satisfaction. Thermal printers still live. They work on the basis of producing type and graphics through heating the paper on a selected dot basis.

Their problems and their virtues have not changed. They are silent, which is their big plus. They need special paper which reacts to heat and is more expensive than ordinary paper – upwards of $2.50 a roll – which is their big minus.

Useless for writing letters on, marvellous for doing quick print-outs of programs and inter-office memos which have a limited life span. (It is possible to make the image on thermal paper permanent by means of photocopying it. This is a system I used for years until I worked out a proper costing and realised it was wildly extravagant).

Internal memos

OTC, who are responsible for telecommunications from Australia to the rest of the world, use them a lot in their internal office memo system.

It is of more than passing interest that OTC, instead of transmitting information from office to office by Local Area Network – which you would confidently imagine would be the cheapest and quickest method – actually send their interoffice memos on their Mercury system via the Midas satellite and can show that this is cost efficient.

For that sort of inter-office communication use, thermal printers are excellent, principally because they are silent and do not wake the over-stressed executive when he is having a well earned siesta after lunch.

But just as we finish writing this, we hear from Apple that they have a heat sensitive printer that works on plain paper.

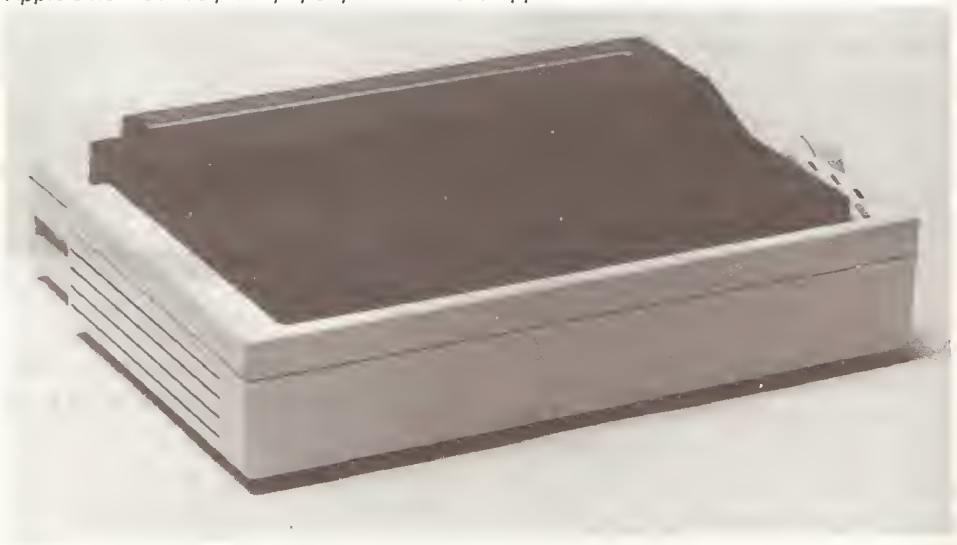
It turns out that this is a bit of a false lead. As soon as I saw the word thermal I leaped to the conclusion that the technology had something to do with the Silentwriter. In fairness, so did some of the executives of Apple Australia.

Now that I have played with the Scribe I find that, yes, it uses thermal transfer but not in the way I thought.

Hot dot matrix

The easiest way to think of it is as a dot matrix printer – of which more in a moment – which uses heated pins onto a ribbon to deposit a wax image on to plain paper. At least, that is how I think it works and nobody has yet told me otherwise. Publicity hand outs have claimed that it is as "quiet

Apple's new Scribe plain paper printer for the Apple IIc



as a whisper". No, it's not. But it is far faster than any other comparable printer (but not quieter than the Silentwriter). All of the noise comes when the head bounces back on the carriage return. When it is traversing the paper it is so quiet you are tempted to lift the lid just to see if an image is being printed on the paper.

It comes with two speeds. Draft at 80 characters per second, better quality at 50 characters per second. Because it is, in effect and in reality, a graphics printer, there is effectively no limit to the number of type faces you can use, although certain typefaces which are built in are more suited to the macjhine's capabilities.

The Scribe is designed by the people who designed the IIc and is in the same off white finish. It decorates a desk rather than detracts from it, and you could quite happily work at your computer while the Scribe prints out if you have a buffer. This printer does not appear to have a buffer of any size built in, which with a quiet printer like this would seem to me to be an absolute necessity.

(Next month we will be testing and reviewing an Australian made and designed buffer with 64K of memory which is a good thing. With a buffer like that you will never need to hang around waiting for the printer to finish).

In typing mode the Scribe works on a 9 x 7 matrix which gives pretty high resolution. I don't want to get into an argument here about what is typewriter quality, but this machine can print out letters which you will not be ashamed to send to your bank manager.

Colours galore

The Scribe can be either a colour printer, using tri-colour ribbons which on a mix and match basis will give you seven colours, or can be used with an all black ribbon. Here is where we come up against the only snag involved with this printer. The ribbons are one time use only. In fact, we have seen samples where the ribbon has been wound back three times and the results were still very acceptable. But the intention is that the ribbons should be one time use only. Therefore this printer is not going to be ideal if you use printers pretty much non-stop. In this publishing office we switch on two of our printers when we get in and they don't seem to stop working until we knock off in the early evening.

The Scribe was not designed for that sort of use. It was designed with the personal user very much in mind. Someone who sends off a few letters a week, but also does graphics and wants to be able to use colour. For that sort of purpose the Scribe is ideal.

The amazing news is that the Scribe, which should be on sale by the time you read this, will be available for something around $500. Which is a small miracle.

Let us now consider converted electric typewriters and get them out of the way. They do not work.

Please accept the fact that it is almost impossible to connect a personal computer to an electric typewriter and to run that set-up for any length of time.

Yes, it has been and is done. But it does not work – not for long. Electric typewriters were simply not designed for the steady pounding they get as copy comes from the computer. Especially in an office. I have never come across an authentic case of one working happily and contentedly – but, as always, I am open for correction. Until someone proves otherwise I sincerely recommend you give them a miss.

Electronic typewriters

Do not confuse electric typewriters with electronic typewriters, many of which – Olivetti, Olympia and Brother – are specifically designed to have a dual life and can be used either as a typewriter or as a printer.

These are often the ideal answer if you need a printer for correspondence and also need to do some quick typing from time to time. They are not designed, and are not suitable for, non-stop pounding. But they produce beautifully typed letters relatively quietly. Secretaries love them. For example, I use an Olympia linked to my Apple IIe, and very satisfactory it is too.

The link between my typewriter and my Apple was designed and made by Olympia in Australia, and the machine types out letters of excellent quality, if slightly slowly compared to other machines.

Next come the dinosaurs – the daisy wheel printers. These print in much the same way but usually faster and noisier. They do not normally have a keyboard, although there are exceptions like the Qume in our office.

Noise and price are the great problems with these printers. One of the answers is the use of acoustic hoods, but these are bulky, comparatively expensive and can interfere with the way that you work.

And I have found daisy wheel printers less than reliable. Indeed, in the office with one of our machines we found that choking it down to half speed was the only way we could get it to operate without constant calls to the service mechanic.

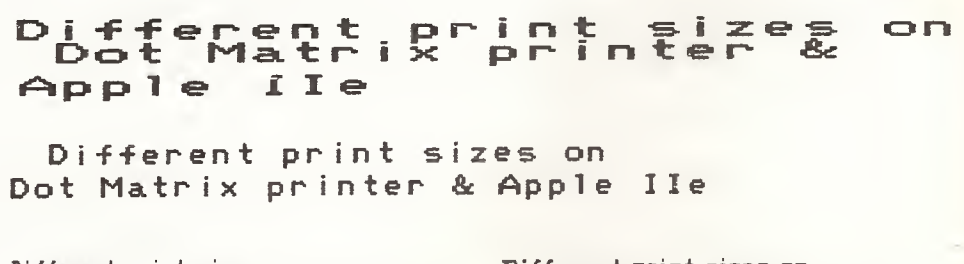
Dying like the dodo

In my opinion daisy wheels are headed the way of the dodo and will soon be extinct. This will endear me not to daisy wheel manufacturers. But somebody has to tell the emperor he is not wearing any clothes.

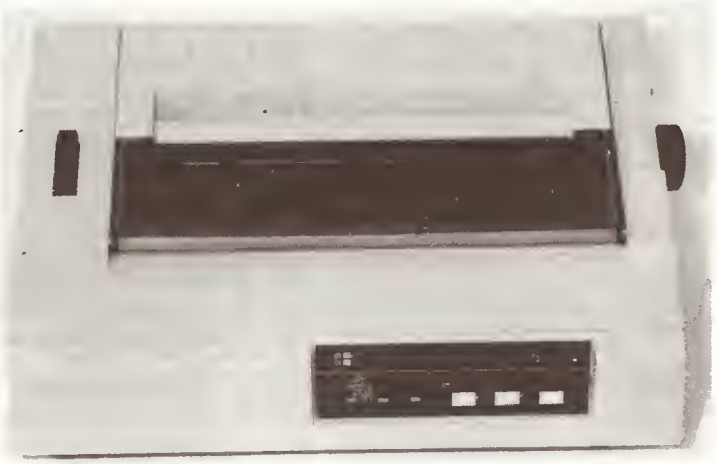
Then there are dot matrix printers.

They used to be cheap and cheerful and the print style they provided looked exactly what it was – a string of dots linked together in the shape of a letter.

This has improved tremendously in the last few years and now some dot matrix printers produce copy which is almost, not quite, up to the standard of the daisy wheel printer. In fact, the Apple Imagewriter is in my



Olympia ES100 electronic typewriter



C.Itoh M8510 colour printer

opinion capable of producing work of such a high quality that it would be a nit-picker who said it was not suitable for correspondence.

Dot matrix printers have an advantage in that they are not tied to any particular type style. The dots are patterned through signals sent from the computer. The number of type styles you can use is, effectively, limitless. However, the number of type styles that look acceptable is rather more limited. Where dot matrix printers miss out is in proportional spacing. I have yet to see proportional spacing effected on a dot matrix printer which is up to my, admittedly high, standards. I cannot for the life of me see why this is. It can't be a killer of a program to write. And it would improve the quality of the output by a very large margin. No doubt we will see this improvement in the near future.

Into colour

To slightly complicate matters, the dot matrix has now gone into colour, which will be of relevance if you produce graphs or other documents which need colour to get their point across and if you are not using the printer full blast all day. If you are, the cost of the ribbons will become prohibitive.

So far they are few and far between and they are being upstaged by the Scribe, as discussed earlier. For example, the C. Itoh 8510C is priced some considerable way over the $1,000 mark and works on the principle of using four colour ribbons, producing colour typing with consummate ease. But it does nothing at that price that the Scribe doesn't do for around $500, so I can't see it being a starter for Apple enthusiasts. Hewlett Packard have got a colour dot matrix printer which I understand will be well under the $1,000 mark and I hope to test it before the next issue.

The laser

Coming up fast over the horizon is the laser printer. We have seen a demonstration of one by Rank Xerox which is the state of the art. But expensive.

However, we will see laser printers from Apple Australia this year – probably October/November – that will sell for under $6,000. This is my forecast and not the official word from Apple. However, they do admit that a laser printer is in the wings. Information from other sources suggests the price and the timing.

The great advantage of the laser printer is that it operates at speeds vastly exceeding that of the dot matrix but with the quality of the daisy wheel. It is relatively silent and will use any reasonable quality bond paper.

In some cases the paper can be fed in by sheet, and the system used to feed the sheets in – based on the principle used by the Heidelberg printing machine – appears to obviate any paper jams.

When these printers appear we would like to use them to set the type for this magazine. Because they can produce type of a quality that rivals any phototypesetter. We would need a range of types to achieve this, but we know that it is more than theoretically possible.

Ink jets

Then there is the ink jet printer. When I started writing this article I was going to introduce a throwaway line saying that ink jet printers were not suitable for Apple computers. And at the time, last week, I would have been right. Now, Hewlett-Packard have announced an ink-jet printer at well under $1,000 and we will be testing it for the next issue. It is, so I have been told, almost totally silent and very compact.

I am also assured by an ex-journo who works at Hewlett Packard that the machine prints to typesetting bromide quality. Knowing that Hewlett-Packard have never let a badly designed or built piece of equipment come out of the factory, I am willing to believe him.

Big test coming up next month.

All these machines will see the death of the daisy wheel printer in a relatively short period of time – especially in the higher price brackets. The daisy wheel will be squeezed in two directions.

From above, by laser printers which will compete on price and be far superior in terms of quietness, quality and speed.

From below by dot matrix printers which are getting cheaper, faster and better all the time.

From one side by multi-coloured machines like the Scribe.

And from the other by ink jet printers by Hewlett Packard.

That is the news as of this moment. Next month undoubtedly the scene will have changed and there will be full tests of at least two, and possibly three, new printers.

These are exciting times we live in. □

Keeping your Apple cool, clean and carefree

by Gareth Powell

When I was a young lad and there were wolves in Wales my Apple started to act up and I took it into Computerland in Sydney. There behind the service counter was the hawk-visaged Felix Marci, who explained to me that my problem lay in getting hot and bothered. Being Welsh, I am noted for my natural phlegm and sanguine attitude towards life, and I took exception to this remark. Felix explained it was not me that was getting hot and bothered, it was the computer.

I was using too many cards and restricting airflow.

Being always willing to give a new idea a try I took a large hacksaw, cut a quite nasty hole into the side of my Apple and installed a fan. And it turned out that Felix was right. My problem was indeed overheating. Since that day I have modified every Apple I have in the company to have either a fan or fans plus, sometimes, extra ventilation slots.

Heat was not the only problem I was to encounter.

Since those distant days of my departed youth I have learned that the four major rules of computer care are: keep it cool, keep it clean, keep magnets away from it and avoid static electricity by not wearing nylon knickers. This is over-simplifying the situation, of course, but if you follow these rules you will avoid most problems.

Keep it cool

Australia can be a hot place. Temperatures can get tropical. Computers do not like the heat. Indeed, many computers when they get hot and bothered simply refuse to work at all.

This is especially true of most Apples – not the IIc for reasons which we will explain in a moment.

First, the power supply warms up. Then the boards warm up, then the inside of your computer is a hot house, which is not a good thing.

When you have trouble with your Apple the first problem to look for is a heat build-up.

As I write this the temperature is low enough for me to need my thick fisherman's cardigan and my uggie boots, so I know the Apple I am using is contented. In the summer the situation would be altogether different.

Then I would have at least one and sometimes two fans going full blast inside the Apple, sucking in air at one end and blowing it out at the other. Because otherwise the Apple would get overheated and sulk.

Why is this?

Most Apples were not designed for continuous use under arduous conditions. Especially if they have been heavily modified – as my machines have – to expand the memory and increase the speed of operation.

To cope with this extra heat many programmers leave the covers off their Apples, to let the warmth dissipate naturally. This is not a good idea, as in a moment of excitement at finding the secret of the universe and everything else you can drop your morning coffee into the works – not a recommended practice.

CSIRO recommend that you fit two fans to an Apple if you use more than four boards. You can buy fit-on fans of assorted shapes and sizes or you can fit your own. Most dealers seem to be happy to fit one for around $70.

Please note these problems only apply if you have extra boards and the weather is relatively warm. I have known people who have run their Apples without any problems whatsoever. The heat problem in the IIe is less than in the II–, mainly I think because the 80 column board is smaller and cooler. On the IIc there is no problem whatsoever as the chips are CMOS based and their heat output is far less.

If you have an Apple II+, and it starts acting up during the summer, always suspect over-heating first.

Keep it clean

Nothing damages disks and machines as fast as cigarette smoke. Don't smoke near your Apple, and don't let anyone else do so.

Computer disks are platters of thin plastic covered with rust. Your everyday common or garden rust which is ruining your lawn mower.

Avoid static electricity

Static electricity can bomb an Apple out just like that. Being of a nocturnal habit, I do most of my writing at night. I used to wear a blue nylon dressing gown which I discarded when the static electricity it built up shot from me into the Apple and zapped two chips. I actually saw the electricity and thought it was a modern version of St Elmo's fire. Quite eerie.

Since then I've become intensely interested in the problems of static electricity. Computer chips get sick and die at the sight of it.

Honeywell recently produced a study to show that problems caused by static electricity made more computers break down than mechanical and power failures. Static electricity can be caused in many ways, but with personal Apples the problem is, appropriately, people. Walking around you can build up a charge which will typically be around 7,000 volts but can get as high as 10,000 volts. The drier the day the more likely it is you will build up a large static charge. Humidity reduces the problem.

When you learn that anything over 200 volts can damage Apple circuitry you realise the extent of the problem.

Static electricity can not only zap your chips so that the Apple doesn't work, it can also change the information within a chip so that the problem may not show up for months afterwards. Static electricity can make circuits lose memory, affect data that you have entered and mess around with a printout.

Static electricity is to Apples what

the demon booze is to the brain. Only faster.

The damage occurs when you pick up a charge and earth yourself through the Apple.

It was once explained to me that the ideal place to handle chips would be sitting in the middle of a field with one foot in a bucket of water on a dampish day. This is not always convenient.

The molecules of rust – the manufacturers call it iron oxide but it is still rust – have their polarity changed when information is recorded, just like a tape recorder. In order to be able to cram on more and more information the rust is being made into finer and finer particles. Some manufacturers, Nashua are a good example, are now getting them to stand to attention like soldiers on parade. Imagine the problems created by microscopic smoke particles landing on these disk surfaces.

We suffered from an immense number of disk problems in my office until we totally banned smoking. End of disk problems.

Some members of staff have since been sneaking a fag when I am out. Disk problems have started to come back. You cannot smoke near an Apple and expect it to perform properly.

Nor do these machines like dust.

When an Apple is not in use it should be covered. In the United States they have a greater respect for Apples than we have here, and machines are covered and disks are put into boxes when closing down after every session. Not in Australia. You see disks scattered on desks – and they are not even in their envelopes. From such dirty habits come disk crashes and data losses.

The head that picks up the information from the disk is very similar to a tape recorder head and should, in exactly the same manner, be cleaned. One way of doing it is to dismantle the whole disk drive and use cotton buds and alcohol. Which is how I do it. If that is too much of a hassle there are several cleaning disks available.

Rexel supply theirs sealed in a package so you can be sure that after use you have an ultra clean head. How often you perform this chore depends on how often you use the machine. I do it weekly. You may only need to do it once a month.

Keep magnets away from it

If you put a magnet near a disk you wipe it clean. Indeed, this is the way that many commercial companies recycle their disks after use. The biggest culprits are paperclips. In many cases these are kept in a plastic container that you shake like a salt cellar to get the clips to protrude through the top aperture. When you do this you are sending them up to a magnet and magnetising them. Place one of these clips on the top of the disk and the disk is ruined.

If you are using an Apple and one of these paperclip dispensing devices, get rid of one or the other.

You may also find your office scissors have become magnetised, so keep them well away from your disks.

I have also abandoned the idea of permanently earthing myself by trailing a piece of copper wire on the ground. The suggestions regarding where I should stick one end of the wire have been vulgar and not, in my opinion, funny.

The bad effects of static electricity can be minimised during dry spells – the problem is not so bad when it is raining or there is high humidity –by not having your Apple in a carpeted room, by making sure the wiring is properly earthed and by not wearing clothes made from artificial fabrics.

Far better is to use an anti-static mat.

Rexel import one which costs something under a hundred dollars. This is certainly a good investment in an office where an Apple breakdown can leave you floundering at considerable cost in time and money. The operator touches the mat before starting and taps it every now and again to get rid of any static build up. The mat has an earth wire which has to have a good earth connnection, not always an easy task in Australian buildings.

Dirt, magnetism and static electricity are an Apple's worst friends.

The best advice I have ever heard is to treat your disks with the same care and attention that you would treat your favourite long playing record and your Apple the way you would treat a top quality camera.

And please don't smoke around the Apple. If the state of your lungs doesn't worry you, think of the health of your disks. They certainly aren't covered by Medicare. ◘

TIPS AND TECHNIQUES

by Graham Black

RECOVER (from disaster)

Have you ever deleted a program by typing NEW or INT accidentally? It happens to the best of us at the most inconvenient times. Here is a good little machine language program that will restore it for you. To use it, BRUN RECOVER from your disk directly after your accident. When you type LIST your old program will be back again.

NOTES:

1. Only APPLESOFT programs can be recovered.
2. If you typed INT you must type FP to return to APPLESOFT before BRUNing RECOVER.
3. To execute the program just BRUN RECOVER or BLOAD RECOVER and CALL 768.

```
0300- A2 00 86 06 A2 08 86 07
0308- A0 05 B1 06 C8 D0 02 E6
0310- 07 C9 00 D0 F5 8D 00 08
0318- 8C 01 08 A6 07 8E 02 08
0320- A2 02 D0 02 A2 03 B1 06
0328- C8 D0 02 E6 07 C9 00 D0
0330- F3 CA D0 F2 84 69 84 AF
0338- A6 07 86 6A 86 B0 60
```

BSAVE RECOVER, A$300, L$3E.

Levels of Apple Computing

by Don Pugh

People have told me that computers are one of the most challenging and frustrating inventions of the twentieth century. As I scratch my head and struggle over some maddening problem, I'm inclined to agree. Here's a simple model which has helped me to straighten out my progression in computing.

There appears to be a logical series of steps which almost all users go through in learning how to use their Apple and Apple programs. As an Apple user, it's useful to know where one's at in order to determine where one should go.

This may be done by finding out your level of use, particularly if the computer is being used to assist others, be it one's children or perhaps the secretary at the office.

By being able to ascertain your friend's level of use of the Apple, it is easier to help by providing relevant information. For instance, your secretary may be struggling at a mechanical level, trying to learn the numerous commands of a new word processing program. She'll probably throw a flower pot at you, or burst into tears and resign, if you provide her with information on some new and more sophisticated program designed to do word processing better. You need to know where she's at in order to help effectively.

I have found satisfaction in reviewing my gradual progress through a series of levels in word processing on my Apple.

Non-use

Until 1980 I had no interest in computers and indeed was not really aware of the micro's uses. Their existence was brought to my attention following an accident in which I became a quadriplegic as a result of a car roll-over. Finding myself in a wheelchair with very little hand movement, I needed to be able to improve my typing skills. Through the help of friends I developed an interest in computing. This led me to the second level of computer use, orientation.

At the orientation level I was interested in seeking out more information concerning computers. At this time I had not yet decided on which computer to buy. After examining the advantages and disadvantages of using a cassette operated Pet Commodore, I purchased an Apple on the advice of a friend who was already using one. Back in 1980, when I made this purchase, prices were astronomical. I had a hard time convincing my bank manager to loan me $4,000 for an Apple with two disk drives and a black and white monitor.

I then had the problem of choosing which word processing program to purchase. I didn't know anything about word processing. There were very few articles on word processing and the only programs that I knew about in 1980 were the Apple Writer and Sandy's.

I was sold on Apple Writer simply because it was readily available at my local computer store. Arriving at home with the program and the manual, I was ready to move to the next level of computer use.

At this point I had reached the preparation level. At this level the user is preparing to use his innovation.

Putting a disk into the disk drive, I was pleased to find the Apple Writer Tutorial program which guided me through all the basic stages of this early but useful program. I was delighted. I found that within a couple of hours I had mastered all the basic commands to correct all my typing errors, to move blocks of texts around, or to do word searches.

With some extra work I learned about text embedded commands and various ways of formatting my printing output. I was rapidly moving beyond the preparation level to that of mechanical usage.

Mechanical use

At the mechanical level the user uses the innovation in an uncoordinated manner and makes frequent changes in his usage. I found that to remember the commands and to be able to function effectively I had to use Apple Writer for forty minutes a day for about a week. I found I frequently forgot commands or used the wrong command at the wrong time, wiping out blocks of text.

I was still learning about disk care and found all the problems which one can have if disks are poorly cared for. Slopping port or coffee, leaving disks in the sunshine on the table led to considerable problems.

After a number of disk failures and much profanity, I started becoming very tidy and careful in my care of equipment. Gradually over a month I became familiar with the commands and got my typing speed up to about 10 words a minute, not bad for a one fingered typist. I had moved from the mechanical level to that of the routine level.

Apple IIe. monitor and disk drive

Routine

At the routine level the user makes few or no changes and has established a pattern of use.

In using Apple Writer I found that I maintained routine usage for a period of about two years. I had learned the commands, could use them quickly and efficiently and felt familiar and comfortable with the program. It seemed to do all the jobs that I wanted for word processing.

Nevertheless, after a year I began to encounter a few problems. I really wanted to underline titles, which I did not seem to be able to do with Apple Writer. I also wanted to be able to use two disk drives easily. I wanted to use lower case. Consequently I began asking other users how they dealt with these problems and began to search computer magazines for answers. I was moving beyond the routine level to that of refining my use.

Refinement

At the refinement level the user makes changes to increase his efficiency and output.

In my use of Apple Writer I have found a small program which enabled me to underline titles. This program was easily added to my Apple Writer disk and effectively solved that problem. I also found that by purchasing a lower case chip I obtained lower case for use with Apple Writer. By playing around with the control "D" Apple Writer command I found it was possible to leave my data disk in drive 2 and my Apple Writer disk in drive 1. I had refined the program far beyond its original package format.

I was so excited by these changes that I shared them with other people in the Apple Users Club. This behaviour was leading me into the integration level of use.

Integration

At the integration level the user makes deliberate efforts to co-ordinate with others in using the innovation.

I found that I was led into teaching short courses in word processing using Apple Writer. The program was very useful but I found through my contact with other people that it really did have very serious shortcomings.

I was very impressed by friends who were using 80 columns in their word processing and had such features as micro justification. Apple Writer for me was rapidly losing its appeal. I was moving onto the final level of use, renewal.

At the renewal level the user seeks more effective alternatives to the established use of the innovation.

I found a friend using WordStar. He seemed to be able to do everything that I couldn't with Apple Writer. It was pleasant to see words not split in half. It was nice to see the page formatted the way the printout would appear with the page breaks marked.

At this point I bought myself an 80 column and CP/M card and got into CP/M in a big way. My Apple Writer disks were consigned to the back of the disk box where they have been collecting dust.

Meanwhile I suffered all the anxiety which comes from once again moving myself through the lower levels in my use of WordStar. I am probably now at the routine level although I occasionally refine my usage through discovering new commands. I'm moving towards integration by meeting with friends to discuss difficulties.

I also have some renewal interest when I see some of the newer word processors that present an entire A4 page on the screen or offering a windowing facility involving merging spread sheets and graphics.

Being able to analyse your level of use is a skill which could prove helpful. If you have been sitting at the routine level for a number of years perhaps it's time you thought about doing some research to improve your efficiency. Perhaps you should investigate a new program. By gauging your friend's usage you may also become a better facilitator. Now, how about giving me a hand with dBase II programming. □

Turnover your Apple

Starting with the next issue we are reserving some advertising space for you, our readers.

Our new Apple classifieds will have buy, sell and swap sections, so if you are interested in upgrading your setup, swapping programs or buying something new, this is the place to advertise. Remember that all our readers either own or intend to own Apple equipment.

The rates are reasonable, too. $10 for 30 words, (which works out to about one column inch; easily readable) and 30¢ per word over the 30.

So place your advertisement (send payment and wording to **"Turnover your Apple"**, Top Rear, 4 Carrington Rd, Randwick, 2031) and wait for the calls to come flooding in.

The Australian Apple program contest

We have had several entries worthy of note for our program writing contest. (See Australian Apple Review Vol 1 No 4, April issue.)

From **Douglas Dearinger** in West St Clair NSW comes this neat program which allows you to **design your own Hi-Res picture easily.**

```
]?""

]LIST

5  REM  BY DOUGLAS DEARINGER
6  REM  4/4/1983
10  REM  FOR APPLE//e COMPUTER
20  ONERR  GOTO 1240
30 G$ =  CHR$ (7):D$ = CHR$ (13) + CHR$ (4)
40  TEXT : GOSUB 760
60  HOME
70  INPUT "DOT COLOR = ";C
80  HOME
90  HCOLOR= C
100 PC = C
110  INPUT "STARTING POSTITON (X,Y): ";X,Y
115  HGR : PRINT  PEEK (49234)
120  HPLOT X,Y
130  POKE 34,21
140  HOME : PRINT "X = ";X: PRINT "Y = ";Y
145  NORMAL
150  GET K$
155  IF K$ = "1" THEN  GOSUB 3000
160  IF K$ = "X" THEN  GOSUB 1090
170  IF K$ = "Y" THEN  GOSUB 1120
180  IF K$ = "2" THEN  GOSUB 650
190  IF K$ = "3" THEN  GOSUB 710
200  IF K$ = "4" THEN  GOSUB 570
210  IF K$ = "0" THEN  GOTO 40
220  IF K$ = "5" THEN  GOSUB 410
230  IF K$ = "6" THEN  GOSUB 470
240  IF K$ = "7" THEN  GOSUB 530
245  IF K$ = "8" THEN  GOSUB 4000
250  IF K$ = "I" THEN Y = Y - 1
260  IF K$ = "J" THEN X = X - 1
270  IF K$ = "K" THEN X = X + 1
280  IF K$ = "M" THEN Y = Y + 1
290  IF K$ = "U" THEN X = X - 1:Y = Y - 1
300  IF K$ = "O" THEN X = X + 1:Y = Y - 1
310  IF K$ = "N" THEN X = X - 1:Y = Y + 1
320  IF K$ = "," THEN X = X + 1:Y = Y + 1
330  IF X < 0 THEN  PRINT G$:X = 0
340  IF Y < 0 THEN  PRINT G$:Y = 0
350  IF X > 279 THEN  PRINT G$:X = 279
360  IF Y > 150 THEN  PRINT G$:Y = 150
370  IF K$ = "Q" THEN  GOSUB 1140
380  HCOLOR= PC
390  HPLOT X,Y
400  GOTO 140
410  REM  COLOR CHANGE
420  PRINT "NEW DOT COLOR (0 TO 7):";
430  INPUT PC
440  HCOLOR= PC
450  HOME
460  RETURN
470  INPUT "COLOR TO FILL SCREEN?";SF
480  HCOLOR= SF
490  FOR I = 0 TO 279: HPLOT I,0 TO I,150: NEXT I
500  HOME
510  HCOLOR= PC: RETURN
520  REM  END?
530  INPUT "END? (Y/N) ";EN$
540  IF  LEFT$ (EN$,1) <  > "Y" THEN 140
550  TEXT : CALL  - 1184: END
560  REM  COLOR BORDER
570  HOME : INPUT "COLOR OF BORDER?";BC
580  HCOLOR= BC
590  HPLOT 0,0 TO 279,0 TO 279,150 TO 0,150 TO 0,0
630  HCOLOR= PC: RETURN
640  REM  SAVE PICTURE
650  INPUT "NAME TO SAVE  (RETURN TO QUIT):";SN$
660  IF SN$ = "" THEN  RETURN
670  PRINT D$;"BSAVE ";SN$;",A$2000,L$3FFF"
680  HOME : INVERSE : PRINT " PICTURE SAVED":
      NORMAL : FOR PP
      = 1 TO 2000: NEXT PP: RETURN
690  RETURN
700  REM  LOAD PICTURE
710  INPUT "NAME TO LOAD? (RETURN TO QUIT):";NL$
720  IF NL$ = "" THEN  RETURN
730  PRINT D$;"BLOAD ";NL$
740  RETURN
750  REM  INTRODUCTION
760  HOME
770 :
```

```
780  NORMAL
790  VTAB 12: PRINT "Welcome to HI-RES DRAW.This Program     will enable y
     ou to easily desi9n and    save your Pictures in HI-RES."
800  SPEED= 100: INVERSE : VTAB 5: HTAB 13: PRINT " HI-RES DRAW !"
810  SPEED= 255
820  NORMAL
830  VTAB 20: HTAB 5: PRINT "(PRESS ANY KEY TO CONTINUE)";
840  CALL  - 736
850  RETURN
1080  REM  NEWX
1090  INPUT "NEW X=";X
1100  RETURN
1110  REM  NEW Y
1120  INPUT "NEW Y=";Y
1130  RETURN
1140  REM  DRAW SQUARE
1150  INPUT "COLOR OF SQUARE?";CS
1160  INPUT "WIDTH?";W
1170  INPUT "HEIGHT?";H
1180  HCOLOR= CS
1190  HPLOT X,Y TO X + W,Y
1191  HPLOT X + W,Y TO X + W,Y + H
1192  HPLOT X,Y TO X,Y + H
1193  HPLOT X,Y + H TO X + W,Y + H
1230  HCOLOR= PC: RETURN
1240  INVERSE : PRINT G$;" ERROR#"; PEEK (222);" ON LINE "; PEEK (218) +  PEEK
     (219) * 256;".": FOR PP = 1 TO 2000: NEXT PP: GOTO 145
3000  INPUT "ENTER X :";X1
3005  INPUT "ENTER Y :";Y1
3010  HPLOT X,Y TO X1,Y1
3020 X = X1:Y = Y1
3030  RETURN
4000  INPUT "ENTER COLOR (1-7) :";J1
4005  INPUT "NEW X1,Y1 ";X1,Y1
4020  HCOLOR= J1
4030  FOR T4 = Y TO Y1
4040  HPLOT X,T4 TO X1,T4
4050  NEXT T4
4055  LET X = X1: LET Y = Y1
4060  HCOLOR= PC: RETURN

]LI

]RUN
1) DRAWS A LINE
2) SAVES THE PICTURE
3) LOADS THE PICTURE
4) DRAWS A BORDER
5) CHANGES THE COLOR
6) COLORS THE SCREEN
7) FINISH (END)
8) DRAWS A COLOR BOX
0) RESTART (CLEARS THE SCREEN)

THESE ARE THE INSTRUCTIONS TO USE
HI-RES DRAWING BOARD.THE PROGRAM SAVES
THE PICTURES IN A FILE LOCATED A$2000    L$3FFF
```

A **mailing list program** from **Peter Birthisel** of Warrnambool. He says:

The program is suitable for maintaining a small mailing list using an Apple II+, one disk drive and a suitable printer. Although the concept of such a program is not original it seems to me that these types of programs which are currently available are either much more sophisticated and hence very costly or are provided as part of a word processing package. In either case the person who desires to maintain a small mailing list for a club or association is "over-buying" when in fact he may require only a few simple features.

The program is designed to maintain 100 names and addresses but this can easily be adjusted by changing the DIM command in Line 310. There has been some attempt at modularity and this should be apparent by examining the listing provided. Although the program is fairly lengthy there has been some effort to provide reasonable error-trapping routines so that a first-time user would find it difficult, although certainly not impossible, to "hang" the program.

Most of the listing, particularly to Apple II+ users, should be fairly self-explanatory. In particular notice that CALL-868 (see for example Line 860) is an internal subroutine which clears all text from the current cursor position to the end of the line while maintaining the position of the cursor. A very handy routine to have, particularly when incorrect data is entered and you want to go back to the same screen position without the previous erroneous data remaining there.

```
10  REM       ----------------------
20  REM       +    RANDOM ACCESS    +
30  REM       +      APPLESOFT      +
40  REM       +    MAILING  LIST    +
50  REM       +       DOS 3.3       +
60  REM       +      JAN  1984      +
70  REM       ----------------------
80  GOTO 270
90  REM     --------------------
100  REM        BLANK STRIPPING
110  REM     --------------------
120 I =  LEN (SN$)
130  IF  MID$ (SN$,I,1) =  CHR$ (32) THEN SN$ =  LEFT$
     (SN$,I - 1):I = I - 1: GOTO 130
140 I =  LEN (CN$): IF I = 0 THEN 160
150  IF  MID$ (CN$,I,1) =  CHR$ (32) THEN CN$ =  LEFT$
     (CN$,I - 1):I = I - 1: GOTO 150
160 I =  LEN (A1$)
170  IF  MID$ (A1$,I,1) =  CHR$ (32) THEN A1$ =  LEFT$
     (A1$,I - 1):I = I - 1: GOTO 170
180 I =  LEN (A2$)
190  IF  MID$ (A2$,I,1) =  CHR$ (32) THEN A2$ =  LEFT$
     (A2$,I - 1):I = I - 1: GOTO 190
200  RETURN
210  REM    -------------------
220  REM      ERROR TRAPPING
230  REM    -------------------
240  IF  PEEK (222) <  > 5 THEN 2860
250  POKE 216,0: IF  PEEK (222) = 5 THEN  HOME : VTAB
     10: PRINT "YOU HAVE TRIED TO ACCESS A
     NON-EXISTENT": PRINT "FILE FROM THIS DISK."
260  PRINT D$;"DELETE";F$: HOME : PRINT "THE CURRENT
     DISK HAS THE FOLLOWING FILES": FOR I = 1 TO 1000:
     NEXT I: PRINT D$;"CATALOG": FOR I = 1 TO 8000:
     NEXT I: GOTO 2860
270  REM       --------------
280  REM       INITIALIZATION
290  REM       --------------
300  ONERR  GOTO 210
310  DIM SN$(100),CN$(100),A1$(100),A2$(100),PC$(100)
320 D$ =  CHR$ (4)
330  FOR I = 1 TO 40:UL$ = UL$ + "=": NEXT I
340  HOME
350  VTAB 5: HTAB 8: PRINT "SIMPLIFIED MAILING LIST"
360  HTAB 8: PRINT "========================"
370  VTAB 8: HTAB 10: PRINT "(C)REATE NEW LIST"
380  HTAB 10: PRINT "       OR"
390  HTAB 10: PRINT "(U)SE EXISTING LIST"
400  HTAB 10: PRINT "       OR"
410  HTAB 10: PRINT "(E)ND PROGRAM"
420  GET AN$
430  IF AN$ = "C" THEN  GOSUB 660: GOTO 340
440  IF AN$ = "U" THEN 470
450  IF AN$ = "E" THEN  HOME : END
460  GOTO 420
470  PRINT : PRINT : PRINT
480  INPUT "EXISTING MAILING LIST?  ";F$
490  PRINT D$;"OPEN";F$;",L120"
500  PRINT D$;"READ";F$;",R0"
510  INPUT M
520  PRINT D$
530  REM   ----------
540  REM       MENU
550  REM   ----------
560  POKE 34,0: HOME : HTAB (18): INVERSE : PRINT
     "MENU": NORMAL : PRINT : PRINT
570  PRINT "1---ADD NAMES"
580  PRINT "2---DELETE NAMES"
590  PRINT "3---LIST FILE"
600  PRINT "4---PRINT LABELS"
610  PRINT "5---MODIFY RECORD"
620  PRINT "6---CLOSE CURRENT FILE"
630  PRINT : PRINT : INPUT "SELECTION---";T
640  IF T < 1 OR T > 6 THEN 560
650  ON T GOTO 780,1150,1360,1590,1970,2860
660  REM   --------------------
670  REM     CREATE NEW FILE
680  REM   --------------------
690  PRINT : PRINT : PRINT
700  INPUT "NAME OF NEW FILE?  ";NF$
710  PRINT D$;"OPEN";NF$;",L120"
720  PRINT D$;"DELETE";NF$
730  PRINT D$;"OPEN";NF$;",L120"
740  PRINT D$;"WRITE";NF$;",R0"
750  PRINT 0
760  PRINT D$
770  RETURN
780  REM   -------------
790  REM     ADD NAMES
800  REM   -------------
810  HOME : PRINT : PRINT
820  HTAB (14): INVERSE : PRINT "ADDING NAMES": NORMAL
830  PRINT : PRINT "NO. OF RECORDS CURRENTLY LISTED IS
     ";M
840  PRINT UL$
850  VTAB (22): PRINT UL$;: PRINT "PRESS 'RETURN' TO
     RETURN TO MENU": PRINT "PRESS 'CTRL-E' THEN
     'RETURN' TO CANCEL  CURRENT ENTRY
860  VTAB (10): CALL  - 868: INPUT
     "SURNAME(20)----";SN$
870  IF SN$ = "" THEN 560
880  IF SN$ =  CHR$ (5) THEN 780
890  IF  LEN (SN$) >  = 21 THEN  VTAB 11: CALL  - 868:
     GOTO 860
900  VTAB 21: CALL  - 868: VTAB 11
910  INPUT "FIRST NAME(20)-";CN$
920  IF  LEN (CN$) >  = 21 THEN  VTAB 12: CALL  - 868:
     VTAB 11: CALL  - 868: GOTO 910
930  IF CN$ =  CHR$ (5) THEN 780
940  INPUT "STREET(25)-----";A1$
950  IF  LEN (A1$) >  = 26 THEN  VTAB 13: CALL  - 868:
     VTAB 12: CALL  - 868: GOTO 940
960  IF A1$ =  CHR$ (5) THEN 780
970  INPUT "TOWN/CITY(20)--";A2$
980  IF  LEN (A2$) >  = 21 THEN  VTAB 14: CALL  - 868:
     VTAB 13: CALL  - 868: GOTO 970
```

```
990  IF A2$ =  CHR$ (5) THEN 780
1000  INPUT "POSTCODE(4)----";PC$
1010  IF PC$ =  CHR$ (5) THEN 780
1020  IF  LEN (PC$) <  > 4 THEN  VTAB 15: CALL  - 868:
     VTAB 14: CALL  - 868: GOTO 1000
1030  GOSUB 90: REM  BLANK STRIPPING
1040  PRINT D$;"WRITE";F$;",R";M + 1
1050  PRINT SN$
1060  PRINT CN$
1070  PRINT A1$
1080  PRINT A2$
1090  PRINT PC$
1100 M = M + 1
1110  PRINT D$;"WRITE";F$;",R0
1120  PRINT M
1130  PRINT D$
1140  GOTO 810
1150  REM   ------------------
1160  REM     DELETE RECORDS
1170  REM   ------------------
1180  HOME : HTAB (13): INVERSE : PRINT "DELETING
     NAMES": NORMAL
1190  GOSUB 2230: REM     SEARCH ROUTINE
1200  PRINT : INPUT "DO YOU WISH TO DELETE THIS RECORD?
     ";Z$
1210  IF  LEFT$ (Z$,1) <  > "Y" THEN 560
1220  IF I =  > M THEN 1290
1230  FOR K = I + 1 TO M
1240  PRINT D$;"READ";F$;",R";K
1250  INPUT SN$,CN$,A1$,A2$,PC$
1260  PRINT D$;"WRITE";F$;",R";K - 1
1270  PRINT SN$: PRINT CN$: PRINT A1$: PRINT A2$: PRINT
     PC$
1280  NEXT K
1290 M = M - 1
1300  PRINT D$;"WRITE";F$;",R0"
1310  PRINT M
1320  PRINT D$: PRINT : PRINT
1330  PRINT "THIS RECORD HAS NOW BEEN DELETED.": PRINT
     : INPUT "DO YOU WISH TO DELETE FURTHER RECORDS?
     ";Z$
1340  IF  LEFT$ (Z$,1) = "Y" THEN  POKE 34,0: GOTO 1150
1350  GOTO 560
1360  REM   -------------
1370  REM     LIST FILE
1380  REM   -------------
1390  HOME : HTAB (14): INVERSE : PRINT "LISTING FILE":
     NORMAL : PRINT : PRINT : PRINT
1400  PRINT "READING FILE FROM DISK"
1410  FOR I = 1 TO M
1420  PRINT D$;"READ";F$;",R";I
1430  INPUT SN$(I),CN$(I),A1$(I),A2$(I),PC$(I)
1440  NEXT I
1450  PRINT D$
1460  PRINT : PRINT : PRINT "SORTING FILE INTO
     ALPHABETICAL ORDER"
1470  GOSUB 2700: REM  SORT ROUTINE
1480  HOME : HTAB (14): INVERSE : PRINT "LISTING FILE":
     NORMAL : PRINT : PRINT "MAILING LIST FROM FILE
     ";F$;"": PRINT : PRINT "TO STOP OR RESTART
     LISTING PRESS ANY KEY": FOR I = 1 TO 40: PRINT
     "=";: NEXT I: POKE 34,6: PRINT
1490  FOR I = 1 TO M: IF  PEEK ( - 16384) > 127 THEN
     POKE  - 16368,0: WAIT  - 16384,128,0: POKE  -
     16368,0
1500  PRINT CN$(I); SPC( 2);SN$(I)
1510  PRINT A1$(I)
1520  PRINT A2$(I); SPC( 3);PC$(I)
1530  PRINT : PRINT
1540  FOR A = 1 TO 1000: NEXT A
1550  NEXT I: PRINT "PRESS ANY KEY TO RETURN TO MENU"
1560  GET A$
1570  POKE 34,0
1580  GOTO 560
1590  REM   ----------------
1600  REM     PRINT LABELS
1610  REM   ----------------
1620  HOME : HTAB 14: INVERSE : PRINT "PRINT LABELS":
     NORMAL
1630  PRINT : PRINT
1640  PRINT "WHEN USING DOUBLE WIDTH LABEL PAPER
     ON-SCREEN FORMATTING WILL BE INCORRECT.
1650  PRINT : PRINT "USE THE MENU COMMAND '3--- LIST
     FILE' TO OBTAIN A CORRECT ON-SCREEN LISTING."
1660  FOR I = 1 TO M
1670  PRINT D$;"READ";F$;",R";I
1680  INPUT SN$(I),CN$(I),A1$(I),A2$(I),PC$(I)
1690  NEXT I
1700  PRINT D$
1710  PRINT : PRINT
1720  INPUT "DO YOU WANT A SORTED LIST? ";T$
1730  IF  LEFT$ (T$,1) = "N" THEN 1750
1740  GOSUB 2700
1750  PRINT : INPUT "(S)INGLE OR (D)OUBLE LABEL PAPER?
     ";T$: IF T$ = "D" THEN 1820
1760 N = 1
1770  PRINT D$;"PR# 1": PRINT  TAB( 15);CN$(N); SPC(
     3);SN$(N)
1780  PRINT  TAB( 15);A1$(N)
1790  PRINT  TAB( 15);A2$(N); SPC( 4);PC$(N)
1800  PRINT : PRINT : PRINT : PRINT : PRINT : PRINT :N
     = N + 1: IF N > M THEN 1950
1810  GOTO 1770
1820  PRINT D$;"PR# 1"
1830 N = 1
1840  PRINT  TAB( 4);CN$(N); SPC( 3);SN$(N); TAB(
     50);CN$(N + 1); SPC( 3);SN$(N + 1)
1850  PRINT  TAB( 4);A1$(N); TAB( 50);A1$(N + 1)
1860  PRINT  TAB( 4);A2$(N); SPC( 4);PC$(N); TAB(
     50);A2$(N + 1); SPC( 4);PC$(N + 1)
1870  PRINT : PRINT : PRINT : PRINT : PRINT : PRINT
1880 N = N + 2
1890  IF N = M THEN 1920
1900  IF N > M THEN 1950
1910  GOTO 1840
1920  PRINT  TAB( 4);CN$(N); SPC( 3);SN$(N)
1930  PRINT  TAB( 4);A1$(N)
1940  PRINT  TAB( 4);A2$(N); SPC( 4);PC$(N)
1950  PRINT D$;"PR#0"
1960  GOTO 560
1970  REM   ------------------
1980  REM     MODIFY RECORDS
1990  REM   ------------------
2000  HOME : HTAB (12): INVERSE : PRINT "MODIFYING
     RECORD": NORMAL
2010  GOSUB 2230: REM  SEARCH ROUTINE
2020  PRINT : INPUT "DO YOU WANT TO CHANGE THIS RECORD?
     ";Z$: PRINT
2030  IF  LEFT$ (Z$,1) <  > "Y" THEN 560
2040  PRINT : INPUT "WHICH RECORD IS WRONG? ";NB
2050  IF NB < 1 OR NB > 5 THEN  PRINT : HTAB 10: FLASH
     : PRINT "ALTERNATIVES ARE 1-5": NORMAL : PRINT :
     GOTO 2040
2060  PRINT : PRINT "TYPE IN CORRECT RECORD.";NB;:
     INPUT "---";CR$(NB)
2070  IF NB = 1 THEN SN$ = CR$(NB): IF  LEN (SN$) > 20
     THEN  VTAB 22: CALL  - 868: VTAB 23: CALL  - 868:
     GOTO 2060
2080  IF NB = 2 THEN CN$ = CR$(NB): IF  LEN (CN$) > 20
     THEN  VTAB 22: CALL  - 868: VTAB 23: CALL  - 868:
     GOTO 2060
2090  IF NB = 3 THEN A1$ = CR$(NB): IF  LEN (A1$) > 25
     THEN  VTAB 22: CALL  - 868: VTAB 23: CALL  - 868:
     GOTO 2060
2100  IF NB = 4 THEN A2$ = CR$(NB): IF  LEN (A2$) > 20
     THEN  VTAB 22: CALL  - 868: VTAB 23: CALL  - 868:
     GOTO 2060
2110  IF NB = 5 THEN PC$ = CR$(NB): IF  LEN (PC$) <  >
     4 THEN  VTAB 22: CALL  - 868: VTAB 23: CALL  -
     868: GOTO 2060
2120  PRINT : INPUT "ANY MORE CORRECTIONS TO THIS
     RECORD? ";Z$: PRINT
2130  IF  LEFT$ (Z$,1) = "Y" THEN 2040
2140  POKE 34,0
2150  PRINT D$;"WRITE";F$;",R";I
2160  PRINT SN$
2170  PRINT CN$
2180  PRINT A1$
2190  PRINT A2$
2200  PRINT PC$
```

```
2210  PRINT D$
2220  GOTO 560
2230  REM     --------------------
2240  REM       SEARCH  ROUTINE
2250  REM     --------------------
2260  VTAB 7: PRINT UL$: VTAB 19: PRINT UL$: POKE 34,19
2270  VTAB 21: PRINT "     FOR SEARCH ON SURNAME ONLY
     PRESS": PRINT "     'RETURN' INSTEAD OF FIRST
     NAME."
2280  VTAB 3: INPUT "SURNAME OF RECORD(20)--";V$
2290  IF V$ = "" THEN 560
2300 I =  LEN (V$)
2310  IF  MID$ (V$,I,1) =  CHR$ (32) THEN V$ =  LEFT$
     (V$,I - 1):I = I - 1: GOTO 2310
2320  IF  LEN (V$) >  = 21 THEN  VTAB 4: CALL  - 868:
     VTAB 3: CALL  - 868: GOTO 2280
2330  VTAB 5: INPUT "FIRST NAME(20)---------";W$
2340  IF W$ = "" THEN 2460: REM       SEARCH ON SURNAME
     ONLY
2350 I =  LEN (W$)
2360  IF  MID$ (W$,I,1) =  CHR$ (32) THEN W$ =  LEFT$
     (W$,I - 1):I = I - 1: GOTO 2360
2370  IF  LEN (W$) >  = 21 THEN  VTAB 6: CALL  - 868:
     VTAB 5: CALL  - 868: GOTO 2330
2380 I = 1:C = 1
2390  PRINT D$;"READ";F$;",R";I
2400  INPUT SN$,CN$
2410  IF V$ = SN$ AND W$ = CN$ THEN 2540
2420  PRINT D$:
2430 I = I + 1
2440  VTAB 20: CALL  - 958: VTAB 21: IF I > M THEN
     FLASH : PRINT "RECORD NOT FOUND.RETURNING YOU TO
     MENU.": FOR J = 1 TO 2500: NEXT J: NORMAL : GOTO
     560
2450  GOTO 2390
2460  VTAB 21: CALL  - 958:I = 1:C = 2
2470  PRINT D$;"READ";F$;",R";I
2480  INPUT SN$
2490  IF V$ = SN$ THEN 2540
2500  PRINT D$
2510 I = I + 1
2520  VTAB 20: CALL  - 958: VTAB 21: IF I > M THEN :
     FLASH : PRINT "RECORD NOT FOUND.RETURNING YOU TO
     MENU.": FOR J = 1 TO 2500: NEXT J: NORMAL : GOTO
     560
2530  GOTO 2470
2540  PRINT D$;"READ";F$;",R";I
2550  INPUT SN$,CN$,A1$,A2$,PC$
2560  PRINT D$: FOR AB = 10 TO 16: VTAB AB: CALL  -
     868: NEXT AB
2570  VTAB 10: PRINT "RECORD FOUND"
2580  PRINT "============"
2590  PRINT "1-SURNAME-----";SN$
2600  PRINT "2-FIRST NAME--";CN$
2610  PRINT "3-STREET------";A1$
2620  PRINT "4-TOWN/CITY---";A2$
2630  PRINT "5-POSTCODE----";PC$
2640  HOME : VTAB 21
2650  INPUT "IS THIS THE RECORD YOU REQUIRE? ";Z$
2660  IF  LEFT$ (Z$,1) = "Y" THEN  RETURN
2670  PRINT : INPUT "DO YOU WISH TO CONTINUE SEARCHING?
     ";Z$
2680  IF  LEFT$ (Z$,1) <  > "Y" THEN 560
2690  ON C GOTO 2430,2510
2700  REM     --------------------
2710  REM       SORT  SUBROUTINE
2720  REM     --------------------
2730 D = 2 ^  INT ( LOG (M) /  LOG (2)) - 1
2740  FOR I = 1 TO M - D
2750  FOR J = I TO 1 STEP  - D
2760  IF SN$(J) <  = SN$(J + D) THEN 2830
2770 Z$ = SN$(J):SN$(J) = SN$(J + D):SN$(J + D) = Z$
2780 Z$ = CN$(J):CN$(J) = CN$(J + D):CN$(J + D) = Z$
2790 Z$ = A1$(J):A1$(J) = A1$(J + D):A1$(J + D) = Z$
2800 Z$ = A2$(J):A2$(J) = A2$(J + D):A2$(J + D) = Z$
2810 Z$ = PC$(J):PC$(J) = PC$(J + D):PC$(J + D) = Z$
2820  NEXT J
2830  NEXT I
2840 D =  INT (D / 2): IF D > 0 THEN 2740
2850  RETURN
2860  REM     --------------
2870  REM       CLOSE FILE
2880  REM     --------------
2890  PRINT D$;"CLOSE"
2900  GOTO 340
```

TIPS AND TECHNIQUES

CHANGING PASSWORD

```
10    REM - CHANGING PASSWORD
20    REM
30    REM - THE NUMBER OF PASSWORDS
40    REM - IS SET IN THE FIRST DATA
50    REM - STATEMENT IN LINE 140 AND
60    REM - THE PASSWORDS THEMSELVES
70    REM - ARE IN LINE 160
80    REM
90    REM
100 HOME

110    REM - SET UP ARRAY FOR
115    REM - NUMBER OF PASSWORDS
120 DIM PW$(11)

130    REM - NUMBER OF PASSWORDS
140 DATA 11

150    REM - PASSWORDS
160 DATA ONE,TWO,THREE,FOUR,FIVE,SIX,SEVEN,EIGHT,NINE,TEN,FINISH

170    REM - READ FIRST DATA ELEMENT
180 READ A

190    REM - READ PASSWORDS AND PUT
200    REM - THEM INTO AN ARRAY
210 FOR I = 1 TO A: READ PW$(I): NEXT I

220    REM - ERAISE 10TH LINE
230 VTAB 10: CALL - 868
240 PRINT "ENTER PASSWORD ";

250    REM - NUMBER OF ATTEMPTS AT PASSWORD
260 FOR P = 1 TO A

270    REM - FIND LENGTH OF PASSWORD
280 FOR I = 1 TO LEN (PW$(P))

290    REM - ENTER A CHARACTER AND PRINT *
300 GET AN$: PRINT "*";

310    REM - IF CHARACTER IS CORRECT
320    REM - GO BACK AND GET THE
325    REM - NEXT CHARACTER
330 IF MID$ (PW$(P),I,1) = AN$ THEN X$ = X$ + AN$: NEXT I

340    REM - IF CHARACTER IS WRONG, EXIT LOOP
350    REM - AND GO TO ERROR ROUTINE
360 IF X$ < > PW$(P) THEN I = LEN (PW$(P)): GOTO 5020

370    REM - PASSWORD IS CORRECT
374    REM - RESET P AND EXIT LOOP
376 P = A: NEXT P
380 PRINT : VTAB 20
390 PRINT "YOU GOT THE CORRECT PASSWORD"

395    REM - END PROGRAM
400 END

5000    REM - ERROR ROUTINE
5010    REM - SET X$ TO NULL STRING
5020 X$ = "": PRINT
5030 VTAB 20

5040    REM - GET NEXT DATA STATEMENT
5050 READ BL$

5060    REM - PRINT TEXT, DATA, AND BELL
5070 PRINT"YOU HAVE BLOWN YOUR ";BL$;" ATTEMPT";CHR$ (7)

5080    REM - PAUSE ROUTINE
5090 FOR F = 1 TO 1000: NEXT F

5100    REM - CLEAR FROM 10TH LINE TO
5105    REM - BOTTOM RIGHT HAND CORNER OF SCREEN
5110 VTAB 10: CALL - 958: HTAB 16

5120    REM - GO BACK AND ATTEMPT NEW PASSWORD
5130 NEXT P
5140    REM - DATA FOR BL$
5150 DATA 1ST,2ND,3RD,4TH,5TH,6TH,7TH,8TH,9TH,10TH,LAST
```

And from **The Baileys** of Cessnock, NSW, a program to **help you win at Lotto**. Their letter says:

Congratulations for taking the plunge and starting a magazine that serves the needs of the Australian Apple users family. We have, in the past, purchased Australian published magazines but found them too widespread (and becoming wider spread with the growth of the computer industry) to actually warrant taking out a subscription. Rather, we purchased the foreign Apple magazines – Softalk and Nibble etc. Keep growing and good luck.

This program was written by a high school member of our family, Greg Bailey, on an Apple IIe.

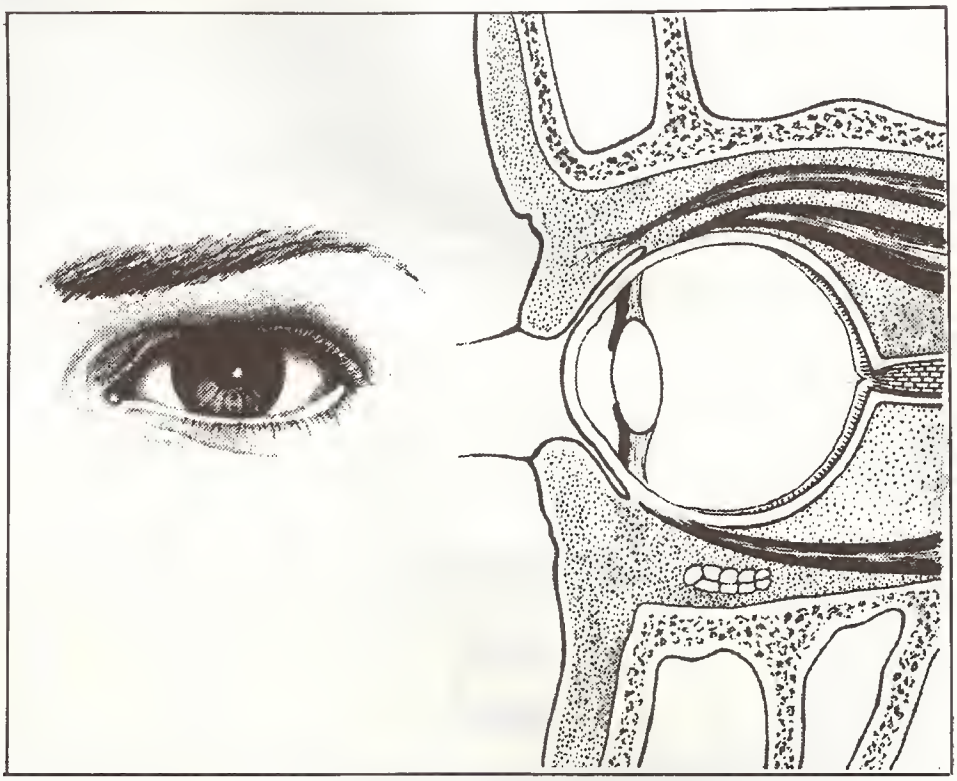
Ed: The Baileys – a family which is plainly into computing – fronted at the office just as this issue was going to press. Sadly, the publisher was tied up in a quite ridiculous and futile business meeting and hardly had time to say "G'day". We would like to meet up with them some time in the near future. □

```
10  REM  ******************
20  REM  * LOTTO SELECTOR *
30  REM  * BY GREG BAILEY *
40  REM  * CESSNOCK HIGH  *
50  REM  * N.S.W.   2325  *
60  REM  ******************
70  REM  ** DRAW NUMBERS **
75  DIM A(16)
80  HOME : INPUT "HOW MANY NUMBER
     S TO BE CHOSEN (6-15)";S
85  IF S < 6 OR S > 15 THEN 80
90  HOME : INVERSE : HTAB 12: PRINT
     " LOTTO SELECTOR ": NORMAL
95 X = 12:Y = 8: FOR N = 1 TO 40
100  HTAB X: VTAB Y: PRINT N
110 X = X + 3: IF X = 30 THEN X =
    12
120  IF N = 6 OR N = 12 OR N = 18
     OR N = 24 OR N = 30 OR N =
    36 THEN Y = Y + 2
130  NEXT N
140  REM  ** SELECT NUMBERS **
150  FOR I = 1 TO S
160 A(I) =  INT ( RND (1) * 40) +
    1
170  NEXT I
180  FOR I = 1 TO S: FOR J = I +
    1 TO S + 1
190  IF A(I) = A(J) THEN 150
200  NEXT J: NEXT I
210  REM  ** SHOW NUMBERS **
220 X = 12:Y = 8
230  FOR N = 1 TO 40
240  FOR I = 1 TO S: IF A(I) = N THEN
     INVERSE
250  NEXT I
260  HTAB X: VTAB Y: PRINT N: NORMAL

270 X = X + 3: IF X = 30 THEN X =
    12
280  IF N = 6 OR N = 12 OR N = 18
     OR N = 24 OR N = 30 OR N =
    36 THEN Y = Y + 2
290  NEXT N
300  REM ** ANOTHER SELECTION **
310  VTAB 24: INPUT "ANOTHER  SEL
    ECTION (Y/N) ";A$
320  IF A$ = "Y" THEN 80
330  PRINT : PRINT "GOOD LUCK!!!!
    !!"
```

Are you damaging your eyesight?

by Duncan McCann



There have been stories going around for a long time that staring into a visual display unit (VDU), a cathode ray tube (CRT), a Video Display Terminal (VDT) – pretty much all the same things with different names –could damage your eyesight and general health.

We have heard stories that in Germany a video screen for commercial purposes cannot be sold unless it uses amber rather than green as the basic colour on the screen.

And that in America one major newspaper strike was connected with the introduction of these screens.

These rumours have now culminated in a well-researched and well-written booklet released by the Australian Journalists' Association entitled "VDT, innocuous or insiduous?"

It is written by Tony Webster, and he has gone to great pains to bring us the good oil on the subject, although it must be accepted that his point of view may be slightly biased.

Subjective impressions

Before we look at Tony Webster's work, a few subjective impressions.

As a company we run word processing machines and computers with visual display units.

Sometimes, when we are producing five magazines in one month and editing a book, the machines are in use seven days a week, often ten hours a day. We are therefore prime subjects for damage through any visual display unit radiations.

Now it is true that after a week of steady pounding away our eyes get tired. And we get headaches.

But we believe our eyes would get tired and we would suffer from headaches if we were doing the same amount of work using typewriters. Excessive use of the eyes in this fashion will always lead to headaches and feelings of eye strain. The obvious answer is to cut back on the amount of work we do. We still haven't found the answer to that one.

Years of experience

For many years we have been doing work with banks and airlines.

They make the most extensive use of visual display units, and if there was going to be any trouble one would logically expect it to appear amongst employees in those two areas.

Yet there seems to be a lack of incidence of any serious eye problems – specifically cataracts – amongst staff who are constantly working with visual display units.

We have checked this carefully with a major airline and a major bank.

They are unaware of the problem simply because no complaint ever been registered. This does not prove that it does not exist. Simply that no one seems to be aware of it.

Could it be a form of electronic panic?

Could it be that when a country in Europe recommended all screens be changed from phosphor green to phosphor orange they were giving a knee-jerk reaction to union pressure and not making a decision based on scientific observation?

Very possibly.

No solid evidence

Although Tony Wells has thoroughly researched his book and has written with great restraint, there is still no evidence cited that indicates the radiations from a visual display unit can affect your eyes.

Sixteen cases of cataracts in the eye have allegedly been reported caused by exposure to visual display units. Cataracts normally come with old age or diabetes, and there is frequently a family history of cataracts. When none of these three conditions exist then a search must be made to find out what else might have caused the cataracts.

These sixteen cases are, in the

main, from one doctor, Milton M. Zaret in New York. His evidence is not completely unambiguous, and other experts have challenged it.

In most of the cases it might be possible to say that exposure to a visual display unit caused the cataracts. It might also be possible to say there were other causes.

The evidence is inconclusive.

Eliminate risk

However, no one should risk eye damage appearing at some time in the distant future, no matter how small the risk.

The easiest way to cut down on any possible damage is to have your visual display unit a reasonable distance away from you and keep the light level of the VDU as low as possible.

The latter has two good effects.

Firstly, it stops you permanently marking the screen if you leave it switched on.

Secondly, it cuts down dramatically on the emission level of microwave or ultra violet rays. The trick is to keep the room lighting relatively low and non-glary so the screen appears well lit in comparison.

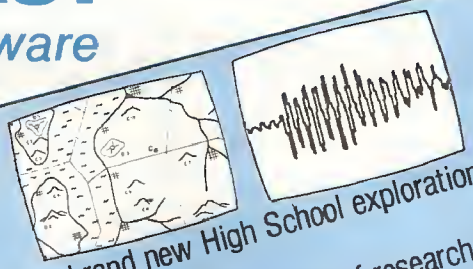
To sum up, on the evidence before us we do not believe visual display units, no matter what they are called, can give you cataracts or otherwise damage your health.

But to be on the safe side seat yourself comfortably, keep the room light down and keep the light emission level of the visual display unit as low as possible consistent with being easily readable with no eye strain.

We will keep a constant watch on this problem because it potentially affects us far more than it does the amateur hacker. We will keep you posted on developments on a regular basis. □

EDUCATIONAL SOFTWARE THAT WORKS!

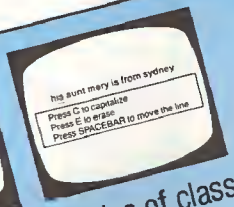
Teacher-approved software for Australian schools



Volcanoes — our brand new High School exploration strategy program for two, three, or four players.

Volcanoes places the student in charge of research teams having to investigate the behaviour of unstable volcanoes in the imaginary province of Wrangelia. Developed by Earthware Computer Services and a professional volcanologist, Volcanoes encourages prudent budgeting of resources (money, equipment, time), the willingness to make decisions based on partial information, and cooperation in dealing with natural hazards.

DLM Arcademic Skill Builders in Maths and Language Arts combine computer game fun with sound educational principles. Each of the twelve programs allows for variable levels of difficulty, speed and running time, and the content is all-Australian. A child never outgrows DLM software! Now available for the Apple II/IIe, Commodore-64, Atari and IBM-PC.



Hartley Courseware — over forty titles of classroom-proven software. Students in classes K-9 will benefit from titles such as Capitalization, Calendar Skills, Clock, Maths Concepts, Verb Usage, Memory Match and Lesson Create. There are even administrative programs such as Parent Reporting.

Hartley Courseware contains Australian material easily tailored by the teacher or parent to particular children's needs — no programming skills are required! Most programs even include on-line record keeping of students progress.

If you have an **Apple II/IIe, Commodore-64, Atari or IBM-PC**, you should take a close look at our computer software. Available from your local computer store and Dominie School Centre, or contact Dataflow for a free colour brochure and the address of a supplier near you.

dataflow

COMPUTER SERVICES PTY LTD 6-8 ELIZABETH ST., BURWOOD. N.S.W. 2134, Ph (02) 745 3303

Software Developer's Conference

Some extracts from a report on the 1st Australasian Software Developers' Conference by **Peter Watson**, which appeared in the May issue of AUSOM NEWS, the newsletter of the Apple Users' Society of Melbourne.

Sub-Directories

One of the advantages of ProDOS is that it allows the use of disk drives other than the old faithful Disk II. For example, the Profile fixed disk may be used, and I would not be surprised if the 3-1/2 inch Sony drives (as used in the Mac and Lisa) are made available for the Apple // series soon. One of the reasons that these (usually larger) types of drives were not implemented under DOS 3.3 is the problems associated with directories.

Directory type files are a new concept to Apple users, although those familiar with UNIX or MS-DOS 2.0 will not be surprised to see them. The directory structure under ProDOS works as follows. There are four blocks on track 0 which correspond to the catalog under DOS 3.3. This "Root Directory", however, will only hold 51 files. More blocks could have been allocated, but where do you draw the line between providing enough directory space for a fixed disk and not wasting space on a diskette? The solution was to implement a version of dynamic directory sizing by using files as extra directory space. The number of entries in a directory file is limited only by the amount of space on a disk, as is the number of directory files.

This overcomes one problem, but there is still another. If a fixed disk contains (say) 2000 files, can you imagine having to scroll though them every time you issue a "CATALOG" command? (Not to mention the performance overhead of searching such a directory each time a file is accessed in a program!) To overcome this problem, the directories are set up in a tree structure and accessed via their file names, or via the volume name for the root directory. To explain further, let me set up an example and use it to introduce some new terms.

Pathnames

We have a volume called "/USER" (all volume names begin with a "/"). The root directory of this disk (named "/USER") contains files "PROG.A" and "PROG.B". To access the file "PROG.A", we refer to it by the name "/USER/PROG.A". That is, we refer to its directory then its filename, separated by a "/". This is the "pathname" of the file. Note that under ProDOS, the term "pathname" is normally used in place of the DOS 3.3 term "filename". To execute "PROG.A", for example, we would enter the command -/USER/PROG.A (remember the "-" command!)

The syntax for file and volume names is more restrictive than under DOS 3.3. A name may consist only of the letters A-Z, the numbers 0-9 and the period ".". The maximum length is fifteen characters.

We decide that we want to keep the data files for this program in a separate directory. First we must issue the "CREATE" command to build a (sub)directory file:-

CREATE DATA.A

This creates a file of type "DIR" and with a pathname of "/USER/DATA.A". We then use "PROG.A" to create a data file called "DATAFILE.1", which is put into the new

Apple II ProDOS Users Kit

directory by using the pathname "/USER/DATA.A/DATAFILE.1". Such pathnames may be up to 64 characters long including the "/" separator (the individual elements of the path being up to 15 characters long).

By the way, the CREATE command can actually create a file of ANY type by using the "TXT" parameter. In fact it MUST be used to "pre-create" a file if you want to BSAVE to a filetype other than "BIN".

At this stage you are probably worrying about wearing out the keyboard with all these long pathnames. Worry no more, the "PREFIX" command is here to save us! By issuing the command:-

```
PREFIX/USER/DATA.A
```

we can now refer to the data file simply by using the pathname "DATAFILE.1". When ProDOS is first executed, it sets the prefix to the name of the volume that it was executed from. This means that a converted DOS 3.3 program will correctly access the root directory automatically. The current prefix can be determined by issuing the "PREFIX" command without any operands. In a BASIC program you would do the following:-

```
10 PRINT CHR$(4)"PREFIX"
20 INPUT PREF$
30 PRINT "The current prefix is "PREF$
```

Now let us assume we have created several more data files and sub-directories. The resulting directory structure might look something like this:-

****some characters used here that the Wang doesn't have****

```
        PROG.A              DATAFILE.1
        (DATA.A (dir) ——— (DATAFILE.2
/USER ic- (PROG.B           DATAFILE.1
        (DATA.B (dir) ——— (DATAFILE.2
```

Notice that although there are apparently duplicate FILEnames for the data files, this is not the case since they actually have different PATHnames (/USER/DATA.A/DATAFILE.1 vs. /USER/DATA.B/DATAFILE.1).

There are several side-effects in the use of sub-directories. Firstly, there is a performance overhead in accessing the files in a sub-directory, although it is not as bad as it would be for a single large directory. It is, however, distinctly noticeable even with a single level sub-directory. This means that files whose performance is critical should be placed in the root (volume) directory. Another side-effect involves the "free space" in a catalog display. No matter which directory is being displayed, the free space always refers to the entire volume. This means that the "blocks used" field at the end of the display does not necessarily reflect the sum of the individual "blocks used" for each file. Lastly, let me give an example of a beneficial side-effect. A number of commercial program disks now contains code to load the correct program to run on either a][+ or a //e. A ProDOS based application program could access several machine specific files simply by specifying the appropriate directory, leaving the individual "filenames" untouched throughout the remainder of the program.

Text Files

Moving on at last, let me speak briefly about the changes to text file handling. The only new command is FLUSH – this forces any unwritten data in a file buffer to be written out to disk. The other changes are mostly related to new or changed use of parameters.

The only change to the OPEN command is the removal of the need to always specify the record length of a random access file. Once a file has been opened with the "R" parameter, ProDOS saves the record length as part of the directory entry and will automatically use it as a default whenever the file is opened. The CLOSE command has not changed.

The READ and WRITE commands have a new parameter, the "F" (field) parameter. Files (random or sequential) consist of fields – strings of characters separated by carriage returns (CHR$(13)). These are numbered from 0 upwards within a sequential file or random access record. The "F" parameter tells ProDOS to skip ahead to a given field.

The meaning of the old "B" parameter has changed in a subtle but significant way. Formerly it would set the file position pointer to an ABSOLUTE position in the file – now it moves the pointer ahead to a position RELATIVE to the current file position.

The position command has also gained the "F" parameter. This command will be pretty much superfluous now since READ and WRITE can use the "F" parameter for similar effect.

Lastly, the APPEND command now works with random access files, since ProDOS keeps proper track of the last byte of a file. This also means the file length in the CATALOG command is now accurate to the byte, not to the sector (if you were lucky)! Speaking of file sizes, the maximum size of a file (apart from the physical limitations of the volume itself) is 16 Mbytes. A limitation in the "end-of-file" pointer is the reason you can't have volume sized files.

External Commands

Anybody who has tried to add or modify BASIC or DOS 3.3 commands knows just how hard it is to maintain some sort of standard procedure that allows you to use more than one modification at a time. Most programs simply assume that they are the only resident of a given "hole" in DOS or I/O vector, and promptly "stomp" on anything that is already there. Apple have now defined a usable, standard mechanism under BASIC for adding new ProDOS commands. That is, commands executable directly in "immediate" mode, or with a prefixed "control-D" during program execution. (This raises an interesting point – ProDOS seems to be able to look directly at the Applesoft commands in the BASIC program, but still requires you to use the "control-D" to signal new or ProDOS commands.)

The method for adding a new command is simply to store the address of your routine in the EXTRNCMD jump vector ($BE06). ProDOS will then pass a command buffer to your routine each time it gets a command. If you wish to act upon the command, you tell ProDOS to return control to your "command processing" routine (after parsing the operands for you, if you wish!) Otherwise you tell ProDOS to continue processing normally (or pass control to the "next" routine, if there is one). The space for your code may be a place such as the old faithful $300, or you may ask ProDOS to allocate you some memory between BASIC.SYSTEM and its file buffers – which should be a fairly safe

place! An example of the type of use external commands might be put to is Apple's new version of APA from the DOS Tool Kit. This now runs under ProDOS – without the "&" in front of all the commands!

Machine Language Interface (MLI)

The MLI is one of the less obvious changes in ProDOS, but it is nonetheless an extremely important one. There used to be two ways to perform disk oriented commands under DOS 3.3. If all of DOS was present, you could imitate a BASIC program and "PRINT" a DOS command through COUT ($FDED). This was acceptable if the DOS command did what you wanted, but often this was not the case – for example, there is no DOS command to read a given sector on a disk. The next step was to call one of the subroutines in DOS 3.3 directly – which became a problem if entry points were moved, as sometimes happened between releases of DOS. In fact, the entry points should have changed more often, but the people at Apple were hamstrung by the fact that so much software would become useless if they changed anything. Most people who have used one of the various "fast" DOS'es have experienced frustration at the number of programs which refuse to work for this very reason.

Eventually Apple "bit the bullet" and revamped the whole shebang. It was an enormous change internally, and most machine language programs will need to be modified – but hopefully it will be the last time that it will ever be necessary. All interfacing to the operating system is now done through a set of standard jump vectors (mostly just one!), in the System Global Page. This interface is known as the MLI (Machine Language Interface). We now no longer care if Apple changes the internal code of ProDOS, as long as we can always call the same place in the same way. As I mentioned in the first article, Apple intends to deliberately change the internals to discourage people who might try (heaven knows why!) to bypass the MLI. By the way, the MLI concept is not new – CP/M has had a similar interface for years.

Apple Macintosh 3½" (hard) floppy disk

The MLI is extremely simple to use, looking something like this:-

```
JSR  MLI       ; at $BF00, a jump vector in the System Global Page
DB   CMDNUM    ; a 1 byte number representing the type of call
DW   PARMLIST  ; the address of the parameter list for the call type
BCS  ERROR     ; a branch to a routine to handle errors
```

There are currently 26 ProDOS call types that I know of, covering such things as creating, opening, reading, writing and closing files, reading and writing specific blocks on a volume, allocating and deallocating interrupt vectors, getting information on volumes and files, and reading the system time.

The parameter list is different for each type of call, although in many cases the same list can be used for related call types with only minor changes (eg READ and WRITE). To help ensure the validity of the parameter list, ProDOS requires the first byte to contain the number of parameters for this call type – get it wrong and get an error! (Actually, ProDOS takes great pains to ensure that it is executing 'real' code – for example, interrupt handlers and clock routines must begin with a $DB (CLD) instruction so ProDOS knows that the vectors were not accidently corrupted!). ProDOS also checks any buffer addresses in the parameter list to ensure you are not trying to write to 'protected' areas such as $100-$1FF, $400-$7FF, $BF00-$BFFF and $D000-$FFFF.

There are some 29 different error codes returned in the A register, with the C flag being sent to indicate that an error occurred. This is a lot better than limited DOS 3.3 indications of what went wrong. (BASIC also has several new error codes, such as "FILE BUSY" – you have tried to OPEN an already OPEN'ed file.)

Actually, if BASIC.SYSTEM is present, you can still use something akin to the old COUT method of issuing commands. You simply stuff a ProDOS command ending in CHR$(13) into the GETLN buffer at $200, and JSR $BE03. If the command is not successful (ie the carry flag is set), then the A register will contain the error code and a JSR $BE0C (PRINTERR) will display the relevant error message.

Miscellaneous

I mentioned previously that ProDOS supports interrupts. One article I have seen pointed out that ProDOS can support them all it likes, but there is still a flaw in the monitor ROMs (the location $45 problem) which will catch you eventually. This is strictly true, but ProDOS minimised the time spent in ROM code as far as possible, and guards location $45 as much as possible otherwise. When executing its own code, ProDOS is in the "language card" and is able to set his own interrupt vector which should eliminate part of the risk altogether. Apple have already announced new ROMs for the //e, and I will be surprised if the interrupt bug has not been fixed – especially since the new mouse for the // series can use interrupts.

If you managed to disconnect DOS 3.3 from the keyboard and/or the screen, then a simple]CALL 1002 (or *3EAG) was normally sufficient to resume relations with DOS. However, if you are unlucky enough to disconnect ProDOS (not an easy thing to do), there is no such call. The nearest thing is a call to $3D0 (which jumps to $BE00) which "cold starts" BASIC – and "NEW"s your program!

A recent "Apple Orchard" magazine presented a patch to the ProDOS FILER program to fix a minor problem. I suggest you check to see whether you have the symptoms first, as my copy of the FILER did not. I think it might be only relevant to very early (ie last year) copies of the program.

The new Apple mouse (I have one and love it!) comes with software written under ProDOS. Among other things, it incudes a separate FORMATTER program for use by MOUSEPAINT. I am sure that it would be easy to modify it to allow it to be BRUN as an 'INIT' utility from other programs, overcoming one of the disadvantages with the lack of an 'INIT'command inside a program. □

Adventurer's Corner

by Ed Mehrtens

Zork II

Zork II is also called the Wizard of Frobozz and is more challenging than Zork I. It may have been written by terrorists as the adventurer needs to know how to use explosives, open locked doors without a key, rob banks and use poisons. As normal there are many puzzles, including a riddle to solve, an unmappable maze and magic to learn. Many of the inhabitants and objects are deadly and the Wizard will make things difficult for you.

Keep good maps but remember that the direction back may not be the exact opposite of that which leads to the room. If you have a problem, see if it is in the following list. If it is then substitute the numbers for words in the table, but remember that not all problems are listed and even those which are listed may only contain a clue rather than being solved in full. Good luck and good adventuring.

I would like to thank the Zork User's group for their information on Zork II.

Treasures and useful objects

(1) A pearl necklace. 37,16,187,169.
(2) A gold key. 26,16,186.
(3) A statue. 37,85,188.
(4) A red sphere. 37,85,189,169.
(5) A violin. 37,16,65,46.
(6) A portrait. 37,16,190.
(7) Moby Ruby. 37,16,191,169.
(8) Zorkmid Bills. 37,16,190.
(9) A blue sphere. 37,16,192,169.
(10) Candy. 37,16,181.
(11) A black sphere. 4,193,29.
(12) A gaudy crown. 37,85,46.
(13) A clear sphere. 37,16,194.
(14) Flathead stamp. 37,85,93.
(15) A gold coin. 26,85,195.

Problems?

(1) Locked doors a problem. 8,16,80,38,55,16,90,67,16,139,72,45, 29,62,4,130,29,107,23,85,146.(2) Is the dragon a problem? 19,29, 76,50,72,16,115,169.
(3) Can't catch a unicorn. 120, 134,11,153,100,35,41,14.
(4) Still can't catch it. 142,16, 150,18,16,94.
(5) Can't get past the Guarded Room. 57,16,110,157,16,124,2,67,16,163,80.
(6) Cerberus killing you. 21,16, 166,117,26,87.
(7) Where is it? 43,16,128.
(8) Still can't find it. 40,16,136's, 171,2,141,16,128,47,40,16,132,72,96, 29.
(9) What is the clay brick? 29, 38,122,162.
(10) Can't solve the riddle. 112, 63,85,52,63.
(11) What is the black string? 29,38,85,155,72,67,23,122,162.
(12) Trouble with the Carousel. 16,138,55,16,104,169,78,29..
(13) What do the spheres do? 69,55,72,16,29,74,174,114,16,148,74, 2,83,26.
(14) What is Grue Repellent? 53,16,144.
(15) You can't wake the princess. 98,180.
(16) What is the Odd Angled Room. 29,38,85,179,102,48,103,183,108,2, 159.
(17) Trouble robbing a bank. 14, 38,66,151.
(18) What does the curtain of light do? 29,71,72,105,152.
(19) How do I get into the vault? 48,12,79,16,25,169.
(20) How do I get out of the vault? 17.
(21) What is the basket and canvas? 85,99,154,111.
(22) Can't read some books. 14, 38,151.
(23) Trouble with a rusty box. 21,162,55,16,118,156,2,67,85,155.
(24) Trouble with a circular room. 29,38,85,52,40,55,72,16,106,2,59,140, 55,72,29.
(25) Can't get back to the circular room. 16,51,11,56,6,44,16,140,176s, 55,1,109,75,42,143.
(26) Can't read the writing on the cakes. 53,54,114,16,158,79,16,181, 169.
(27) Trouble with the pool. 3, 16,28,64,72,176,16,140.
(28) Trapped in a cage. 7,16, 95,72,5,29.
(29) Are buttons killing you? 40, 16,95,72,41,29.
(30) What do the buttons do? 78,16,86.
(31) The square button. 9,49,20, 26,16,86.
(32) The round button. 9,13,20, 26,16,86.
(33) The triangular button. 22,16, 86,2,15,10,65,46,55,16,99,169,182, 113, 41,14,36,13,20.
(34) How do I get the treasures in the Wizard's trophy case? 40,77,171, 2,67,85,101,24.
(35) How do I use the red sphere? 21,29,55,16,119,121.
(36) The blue sphere. 21,29,55, 16,116,121.
(37) The clear sphere. 21,29,55, 16,125,121.
(38) The black sphere. 21,29,55, 16,149.
(39) Trouble in the aquarium. 27,84,2,3,145,177.
(40) What do I do with all the treasures? 57,54,72,16,132.
(41) Problems with the Wizard. 30,16,132,72,31,77,171.
(42) How do I cast spells. 53, 16,129,93,81,33,16,171,2,123,16,24.
(43) Some of the descriptions are nonsense. 29,39,85,126,24,26,29.
(44) What spells are used? 127, 184,135,164,137,172,131,178,133, 101,126,147,185,161,173,165,170, 175,160.
(45) Can't get past the landing. 4,32,60,16,68,171.
(46) I have all the points but have not finished the game. 34,16,92,38,91, 16,58,38,89.
(47) Troubles with the grue. 67, 88,168.
(48) Can't get past the Ice Room. 40,16,94,37,61.
(49) Why does the sword glow. 34,4,70,74,169,82,18,73.

Word Table for Zork II

1 ONE	51 REVERSE
2 AND	52 WELL
3 THROW	53 READ
4 YOU	54 THEM
5 LIFT	55 IN
6 UP	56 GOING
7 TELL	57 GIVE
8 IF	58 DOOR
9 SETS	59 POUR
10 AN	60 POUR
11 OF	61 THERE
12 SOUTH	62 WHEN

13 LOW
14 THAT
15 RELEASES
16 THE
17 NORTH
18 FROM
19 ATTACK
20 SPEED
21 PUT
22 STOPS
23 WITH
24 SPELL
63 "
64 CAKE
65 IRON
66 PERFECTLY
67 USE
68 WIZARD'S
69 LOOKING
70 ARE
71 LEADS
72 TO
73 SOMETHING
74 ONE
25 SMALL
26 ON
27 STAND
28 RED
29 IT
30 ASK
31 TAKE
32 MUST
33 POINT
34 WHEN
35 CAN
36 AT
37 IN
38 IS
39 HAS
40 GET
41 DO
42 LEAVING
43 BEHIND
44 HOWEVER
45 CATCH
46 BOX
47 OR
48 WALK
49 HIGH
50 RUN
75 MOVES
76 THEN
77 HIS
78 CONTROL
79 FROM
80 KEY
81 .
82 AWAY
83 SO
84 BACK
85 A
86 CAROUSEL
87 HIM
88 GRUE
89 VISIBLE
90 LOCK
91 DARK
92 CRYPT
93 BOOK
94 DRAGON
95 ROBOT
96 MOVE
97 DEFLATED
98 KISS
99 LOW
100 BLOOD
101 FILCH
102 DIAMOND
103 SOUTH-EAST
104 MACHINE
105 FOUR
106 BUCKET
107 OUT
108 NORTH-WEST
109 HUNDRED
110 LIZARD
111 BALLOON
112 ANSWER
113 CAUTION
150 PRINCESS
151 UNDER-STANDABLE
152 PLACES
153 ROYAL
154 HOT-AIR
155 FUSE
156 HOLE
157 GUARD
158 FLASK
159 SOUTH-WEST
160 FROBOZZLE
114 THROUGH
115 ICE
116 SAPPHIRE
117 COLLAR
118 OBLONG
119 RUBY
120 ONLY
121 STAND
122 PLASTIC
123 INCANT
124 CANDY
125 DIAMOND
126 FANTASISE
127 FALL
128 MENHIR
129 WHITE
130 PUSH
131 FEAR
132 DEMON
133 FUMBLE
134 THOSE
135 FREEZE
136 WIZARD
137 FIERCE
138 BUTTONS
139 PLACEMAT
140 WATER
141 FLOAT
142 SAVE
143 STRANDED
144 LABEL
145 SOMETHING
146 LETTER-OPENER
147 FIREPROOF
148 BLUE
149 PENTACLE
161 FUDGE
162 EXPLOSIVE
163 GOLD
164 FENCE
165 FREE
166 DOG
167 DANGEROUS
168 REPELLENT
169 ROOM
170 FROBIZZ
171 WAND
172 FERMENT
173 FLUORESCE
174 VIEWS
175 FROBNOID
176 EVAPORATE
177 HEAVY
178 FEEBLE
179 BASEBALL
180 HER
181 POOL
182 ,
183 NORTH-EAST
184 FLOAT
185 FRY
186 UNICORN
187 PEARL
188 CHEST
189 DINGY
190 BANK
191 LAVA
192 DREARY
193 CREATE
194 AQUARIUM
195 LEDGE

□

The CAT's out of the bag!

Dick Smith Electronics presents The CAT – more powerful than an expanded Apple IIe plus compatibility – for a fraction of the cost.

1 It's Apple compatible!

That means that you can use a huge range of programs (approximately 15,000) designed to run on the Apple IIe – without paying the Apple IIe's price! Programs for fun, for learning and for business.
And because you aren't 'locked in' to any one supplier, you can save a fortune: buy your programs from us, or buy them somewhere else if they're cheaper! (You could even buy them from Apple . . .)

2 It's more powerful!

With its very large internal ROM (Read Only Memory) – twice the size of the Apple – and larger, easier-to-use keyboard (81 keys compared to the Apple IIe's 63 keys!), high resolution graphics, 8 colours, 6502A second generation processor, it has the features you need – without paying one cent extra for 'standard' features others charge you a fortune for!

3 It's so easy to use!

Just plug it into a video monitor (or any B & W or colour TV set with a $34.90 modulator) – and you're ready to go computing. You don't need to buy anything else!
But when you want to expand, The CAT is ready for serious computing with a range of top quality add-ons and peripherals all at "value for money" Dick Smith Electronics prices.

4 And it's only $699

The bottom line is always the price! And The CAT wins paws down. Feature for feature it more than stacks up against the Apple IIe, the Microbee, plus any other computer you care to name.
And we don't load up the 'options' either – a fully expanded Apple IIe costs over $2,300 while The CAT, similarly optioned, sells for less than $1,500. Affording the best computer has become a whole lot easier . . . with The CAT, from Dick Smith Electronics.

Basic CAT computer X-7500	$699.00
Disk Drive X-7505	$349.00*
Disk Controller X-7510	$149.00*
Emulator Cartridge X-7530	$99.00
Dual Joysticks X-7520	$34.50
RF Modulator X-7550	$34.90
RS-232C Serial Adaptor X-7515	$129.00
Printer Cable (Centronics) X-7540	$39.00
Filer Utilities Disk with DOS & Man. X-7512	$39.00
High resolution RGB Monitor X-1193	$599.00**
RGB Cable X-7560	$39.00

*Limited Stocks ** Arriving Soon

Dick Smith Electronics

COMPUTERCITY

In 46 Dick Smith Electronics stores throughout Australasia

THE cat
DICK SMITH PERSONAL COMPUTER

Get your FREE cat Information Pack (including comparison brochures)

Name _____

Address _____

P/Code _____

Would you use a CAT computer for:

☐ Business ☐ Education ☐ Fun ☐ (Other)

Post to: Dick Smith Electronics, PO Box 321, North Ryde NSW 2113
or call into your nearest Dick Smith Electronics Centre for a demonstration.

Hey kids, get your free CAT stickers from your nearest Dick Smith Electronics Store. A 775/KT

Introducing a Teacher You Can Give to Your Apple.

For the first time in Australia, quality educational courseware is available for your Apple IIE or your Apple II Plus. Control Data PLATO® is proud to announce that from December 1st, new lessons from the vast PLATO computer based education library will be available for your Apple micro.

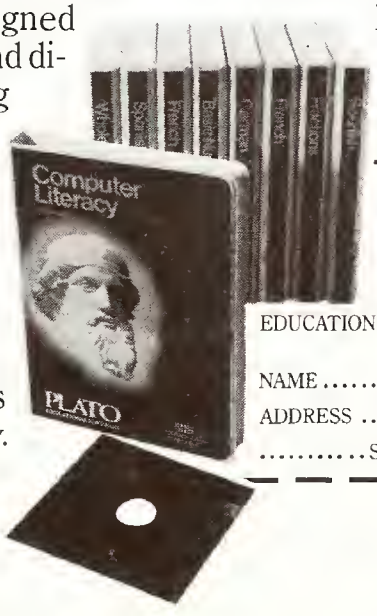
Control Data PLATO computer-based education, recognised as a world leader in educational software, is a highly interactive system of self-paced, one-to-one instruction. Developed over 20 years, it has been thoroughly tested and proven in business, industry, and the academic community. PLATO lessons are designed and programmed to motivate and direct students as well as helping them develop a sense of accomplishment and personal progress.

For example, the course titled Computer Literacy provides the foundation for simple programming as well as giving a brief introduction to the uses of computers in today's society. There are lessons in building vocabulary in French, German and Spanish as well as lessons in Physics, Fractions, Decimals and Basic Number Facts.

Today, more than ever before, all members of society need to learn. PLATO computer-based education is designed and paced for the individual. This means easier learning with infinite patience and personalization.

As traditional education methods become progressively more costly and less efficient you owe it to yourself to bring the power of PLATO learning to your Apple.

If you would like to be on our mailing list for future information on PLATO Courseware for your Apple, please fill in and mail the coupon below.



PLATO
Changing How The World Learns.

Mail to: Marketing Manager,
EDUCATION SERVICES, CONTROL DATA AUSTRALIA PTY. LTD.,
493 St. Kilda Road, Melbourne 3004.

NAME .....
ADDRESS .....
.....STATE ..... POSTCODE .....

CONTROL DATA

Burrows Doble Lawrence CDA/616